

RELAXATIONS OF GF(4)-REPRESENTABLE MATROIDS

BEN CLARK, JAMES OXLEY, AND STEFAN H. M. VAN ZWAM

ABSTRACT. We consider the GF(4)-representable matroids with a circuit-hyperplane such that the matroid obtained by relaxing the circuit-hyperplane is also GF(4)-representable. We characterize the structure of these matroids as an application of structure theorems for the classes of $U_{2,4}$ -fragile and $\{U_{2,5}, U_{3,5}\}$ -fragile matroids. In addition, we characterize the forbidden submatrices in GF(4)-representations of these matroids.

1. INTRODUCTION

Lucas [8] determined the binary matroids that have a circuit-hyperplane whose relaxation yields another binary matroid. Truemper [15], and independently, Oxley and Whittle [12], did the same for ternary matroids. In this paper, we solve the corresponding problem for quaternary matroids. We give both a structural characterization and a characterization in terms of forbidden submatrices.

Truemper [15] used the structure of circuit-hyperplane relaxations of binary and ternary matroids to give new proofs of the excluded-minor characterizations for the classes of binary, ternary, and regular matroids. It is natural to ask if Truemper's techniques can be extended to give excluded-minor characterizations for classes of quaternary matroids. The main results of this paper can be viewed as a first step towards answering this question.

Our structural characterization can be summarized as follows. A matroid has *path width 3* if there is an ordering (e_1, e_2, \dots, e_n) of its ground set such that $\{e_1, e_2, \dots, e_t\}$ is a 3-separating set for all $t \in \{1, 2, \dots, n\}$.

Theorem 1.1. *Let M and M' be GF(4)-representable matroids such that M' is obtained from M by relaxing a circuit-hyperplane. Then M' has path width 3.*

In fact, our main result, Theorem 5.3, describes precisely how the matroids in Theorem 1.1 of path width 3 can be constructed using the *generalized Δ -Y exchange* of [11] and the notion of *gluing a wheel onto a triangle* from [2]. Our description uses the structure of $U_{2,4}$ -fragile matroids from [9] and the structure of $\{U_{2,5}, U_{3,5}\}$ -fragile matroids from [3].

In future work, we hope to obtain a description of these matroids that is independent of the notion of fragility. Specifically, we would like to

characterize the representations of these matroids. As a step in this direction, we describe minimal $\text{GF}(4)$ -representations of matroids with a circuit-hyperplane whose relaxation is not $\text{GF}(4)$ -representable. Note that the proof uses the excluded-minor characterization of the class of $\text{GF}(4)$ -representable matroids. The setup for this result is as follows.

Let M be a $\text{GF}(4)$ -representable matroid on E with a circuit-hyperplane X . Choose $e \in X$ and $f \in E - X$ such that $(X - e) \cup f$ is a basis of M . Then $M = M[I|C]$ for a quaternary matrix C of the following block form.

$$C = \begin{matrix} & \begin{matrix} (E-X)-f & e \end{matrix} \\ \begin{matrix} X-e \\ f \end{matrix} & \begin{bmatrix} A & \underline{1} \\ \underline{1}^T & 0 \end{bmatrix} \end{matrix}.$$

In the above matrix, A is an $(X - e) \times ((E - X) - f)$ matrix, and we have scaled so that every non-zero entry in the row labelled by f and the column labelled by e is 1. Let M' be the matroid obtained from M by relaxing the circuit-hyperplane X . We call the matrix C a *reduced representation* of M . If M' is $\text{GF}(4)$ -representable, then we can find a reduced representation C' of M' in the following block form.

$$C' = \begin{matrix} & \begin{matrix} (E-X)-f & e \end{matrix} \\ \begin{matrix} X-e \\ f \end{matrix} & \begin{bmatrix} A' & \underline{1} \\ \underline{1}^T & \omega \end{bmatrix} \end{matrix}.$$

We have scaled the rows and columns of the matrix such that the entry $C'_{fe} = \omega \in \text{GF}(4) - \{0, 1\}$, and the remaining entries in row f and column e are all 1. The following theorem is our characterization in terms of forbidden submatrices.

Theorem 1.2. *Let M and C be constructed as described above. There is a reduced representation C' of the above form for M' if and only if, up to permuting rows and columns, A and A^T have no submatrix in the following list, where x, y, z denote distinct non-zero elements of $\text{GF}(4)$.*

$$\begin{aligned} & \begin{bmatrix} x & y & z \end{bmatrix}, \begin{bmatrix} x & y \\ 0 & x \end{bmatrix}, \begin{bmatrix} x & y \\ y & x \end{bmatrix}, \begin{bmatrix} x & y \\ z & x \end{bmatrix}, \begin{bmatrix} x & x \\ y & z \end{bmatrix}, \begin{bmatrix} x & x & 0 \\ x & 0 & x \end{bmatrix}, \begin{bmatrix} x & x & 0 \\ x & 0 & y \end{bmatrix}, \\ & \begin{bmatrix} x & x & 0 \\ y & 0 & y \end{bmatrix}, \begin{bmatrix} x & y & 0 \\ x & 0 & y \end{bmatrix}, \begin{bmatrix} x & 0 & 0 \\ 0 & y & z \end{bmatrix}, \begin{bmatrix} x & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & x \end{bmatrix}, \begin{bmatrix} x & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & y \end{bmatrix}, \begin{bmatrix} x & 0 & 0 \\ 0 & y & 0 \\ 0 & 0 & z \end{bmatrix}, \\ & \begin{bmatrix} x & y & x \\ y & y & 0 \\ x & 0 & 0 \end{bmatrix}, \begin{bmatrix} x & y & x \\ y & y & 0 \\ x & 0 & z \end{bmatrix}. \end{aligned}$$

This paper is organized as follows. In the next section, we collect some results on connectivity and circuit-hyperplane relaxation. In Section 3, we prove a fragility theorem. In Section 4, we describe the structure of the

$\{U_{2,5}, U_{3,5}\}$ -fragile matroids. In Section 5, we prove the structural characterization. In Section 6, we reduce the proof of Theorem 1.2 to a finite computer check. This check, carried out using SageMath, can be found in the Appendix.

2. CIRCUIT-HYPERPLANE RELAXATIONS AND CONNECTIVITY

We assume the reader is familiar with the fundamentals of matroid theory. Any undefined matroid terminology will follow Oxley [10]. Let M be a matroid on E , and let $\mathcal{B}(M)$ denote the collection of bases of M . If M has a circuit-hyperplane X , then $\mathcal{B}(M') = \mathcal{B}(M) \cup \{X\}$ is the collection of bases of a matroid M' on E . We say that M' is obtained from M by *relaxing the circuit-hyperplane* X . We list here a number of useful results on circuit-hyperplane relaxation.

Lemma 2.1. [10, Proposition 2.1.7] *If M' is obtained from M by relaxing the circuit-hyperplane X of M , then $(M')^*$ is obtained from M^* by relaxing the circuit-hyperplane $E(M) - X$ of M^* .*

The following elementary results are originally from [7].

Lemma 2.2. [10, Proposition 3.3.5] *Let X be a circuit-hyperplane of a matroid M , and let M' be the matroid obtained from M by relaxing X . When $e \in E(M) - X$,*

- (i) *$M/e = M'/e$ and, unless M has e as a coloop, $M' \setminus e$ is obtained from $M \setminus e$ by relaxing the circuit-hyperplane X of the latter.*

Dually, when $f \in X$,

- (ii) *$M \setminus f = M' \setminus f$ and, unless M has f as a loop, M'/f is obtained from M/f by relaxing the circuit-hyperplane $X - f$ of the latter.*

For a set \mathcal{N} of matroids, we say that a matroid M has an \mathcal{N} -minor if M has an N -minor for some $N \in \mathcal{N}$. We say M is \mathcal{N} -fragile if M has an \mathcal{N} -minor and, for each element e of M , at most one matroid in $\{M \setminus e, M/e\}$ has an \mathcal{N} -minor. We say an element e of an \mathcal{N} -fragile matroid M is *nondeletable* if $M \setminus e$ has no \mathcal{N} -minor; the element e is *noncontractible* if M/e has no \mathcal{N} -minor.

The following lemma is an immediate consequence of Lemma 2.2.

Lemma 2.3. *Let X be a circuit-hyperplane of a matroid M , and let M' be the matroid obtained from M by relaxing X . If \mathcal{N} is a set of matroids such that M' has an \mathcal{N} -minor but M has no \mathcal{N} -minor, then M' is \mathcal{N} -fragile. Moreover, M' has a basis X whose elements are nondeletable such that the elements of the cobasis $E(M') - X$ are noncontractible.*

We use the following connectivity result.

Lemma 2.4. [10, Proposition 8.4.2] *Let M' be a matroid that is obtained by relaxing a circuit-hyperplane of a matroid M . If M is n -connected, then M' is n -connected.*

Kahn [7] proved the following result on the representability of a circuit-hyperplane relaxation.

Lemma 2.5. *Let M' be a matroid that is obtained by relaxing a circuit-hyperplane of a matroid M . If M is connected, then M' is non-binary.*

We use the following definition of the rank function of the 2-sum from [6]. Let M_1 and M_2 be matroids with at least two elements such that $E(M_1) \cap E(M_2) = \{p\}$. Then $M = M_1 \oplus_2 M_2$ has rank function r_M defined for all $A_1 \subseteq E(M_1)$ and $A_2 \subseteq E(M_2)$ by

$$r_M(A_1 \cup A_2) = r_{M_1}(A_1) + r_{M_2}(A_2) - \theta(A_1, A_2) + \theta(\emptyset, \emptyset)$$

where $\theta(X, Y) = 1$ if $r_{M_1}(X \cup p) = r_{M_1}(X)$ and $r_{M_2}(Y \cup p) = r_{M_2}(Y)$, and $\theta(X, Y) = 0$ otherwise.

The next three results on 2-sums and minors of 2-sums are well known.

Lemma 2.6. [10, Proposition 7.1.20] *Let M and N be matroids with at least two elements. Let $E(M) \cap E(N) = \{p\}$ and suppose that neither M nor N has $\{p\}$ as a separator. The set of circuits of $M \oplus_2 N$ is*

$$\mathcal{C}(M \setminus p) \cup \mathcal{C}(N \setminus p) \cup \{(C \cup D) - p : p \in C \in \mathcal{C}(M) \text{ and } p \in D \in \mathcal{C}(N)\}.$$

Lemma 2.7. [10, Theorem 8.3.1] *A connected matroid M is not 3-connected if and only if $M = M_1 \oplus_2 M_2$ for some matroids M_1 and M_2 , each of which has at least three elements and is isomorphic to a proper minor of M .*

Lemma 2.8. [10, Proposition 8.3.5] *Let M, N, M_1, M_2 be matroids such that $M = M_1 \oplus_2 M_2$ and N is 3-connected. If M has an N -minor, then M_1 or M_2 has an N -minor.*

We can now describe the structure of circuit-hyperplanes in matroids of low connectivity. We omit the straightforward proof of the next lemma.

Lemma 2.9. *Let M be a $\text{GF}(4)$ -representable matroid with a circuit-hyperplane H . If M is not connected, then $M \cong U_{1,m} \oplus U_{n-1,n}$ for some positive integers m and n .*

We now work towards a description of the 2-separations of a connected matroid in which the relaxation of some circuit-hyperplane is $\text{GF}(4)$ -representable.

Lemma 2.10. *Let M be a matroid with a circuit-hyperplane X . If A is a non-trivial parallel class of M , then either $A \subseteq E - X$, or $A = X$ and $|A| = 2$.*

Proof. If $A \cap X$ and $A \cap (E - X)$ are both non-empty, then there is a circuit $\{x, y\}$ contained in A such that $x \in X$ and $y \in E - X$. But $E - X$ is a cocircuit of M , so this is a contradiction to orthogonality. Thus either $A \cap X$ or $A \cap (E - X)$ is empty. In the case that $A \cap (E - X)$ is empty, there is a circuit $\{x, y\}$ contained in A that is also contained in the circuit X , so $X = A = \{x, y\}$. \square

For the next result, we say that M is *3-connected up to series and parallel classes* if M is connected and, for any 2-separation (X, Y) of M , either X or Y is a series class or a parallel class.

Lemma 2.11. *Let M be a $\text{GF}(4)$ -representable matroid with a circuit-hyperplane X such that the matroid M' obtained from M is also $\text{GF}(4)$ -representable. If M is connected but not 3-connected, then M is 3-connected up to series and parallel classes.*

Proof. Assume that M has a 2-separation (S, T) where neither side is a series or parallel class. Then M has a 2-sum decomposition of the form $M = N \oplus_2 N'$ for some N and N' with $E(N) \cap E(N') = \{p\}$, where neither N nor N' is a circuit or cocircuit.

First suppose that the circuit X of M has the form $(C \cup C') - p$, where C is a circuit of N , and C' is a circuit of N' while $p \in C \cap C'$. Then

$$(1) \quad r(X) = r(M) - 1,$$

$$(2) \quad r(N) + r(N') - 1 = r(M),$$

and

$$(3) \quad r_M(X) = r_N(C) + r_{N'}(C') - 1.$$

Equation (1) follows from the fact that X is a hyperplane of M ; Equations (2) and (3) follow from the definition of the rank function of the 2-sum of N and N' . Combining (1) and (2), we see that $r(X) = r(N) + r(N') - 2$. Then combining this equation with (3), we see that

$$r(C) + r(C') = r(N) + r(N') - 1.$$

We may therefore assume that C is a spanning circuit of N , and hence that $E(N) = C$ because the hyperplane X is closed. Therefore N is a circuit, a contradiction.

By symmetry, it remains to consider the case when X is a circuit of $N' \setminus p$. Then $r(X) \leq r(N')$. Since X is a hyperplane of M , and $r(M) = r(N) + r(N') - 1$, it follows that $r(N) \leq 2$. Since N is not a cocircuit, we deduce that $r(N) = 2$. Then $r(M) = r(N') - 1$, so $r(X) = r(N') = r(N' \setminus p)$. Since N is not a circuit we deduce that $\text{si}(N) \cong U_{2,m}$ for some $m \geq 4$. Moreover, p is not in a non-trivial parallel class in N otherwise X is not a hyperplane of M .

Switching to M^* , we see that $r_{M^*}(N') = 2$. As above, it follows that $\text{co}(N') \cong U_{n-1,n+1}$ for some $n \geq 3$. Moreover, p is not in a non-trivial series class in N' . Let X_1 consist of one representative of each series class of N' , and let Y_1 consist of one representative of each parallel class of N . By contracting elements of $X - X_1$ and deleting elements of $(E(M) - X) - Y_1$, we obtain $U_{n-1,n+1} \oplus_2 U_{2,m}$ as a minor of M for some $n \geq 3$ and $m \geq 4$. Moreover, X_1 is a circuit-hyperplane of this minor whose relaxation is $\text{GF}(4)$ -representable. Thus $X_1 \subseteq E(U_{n-1,n+1})$. Contract $n - 3$ elements from X_1

and delete $m - 4$ elements from Y_1 to get $U_{2,4} \oplus_2 U_{2,4}$. Relaxing a circuit-hyperplane of this minor gives P_6 which is not $\text{GF}(4)$ -representable (see [10, Proposition 6.5.8]), a contradiction. \square

3. A FRAGILITY THEOREM

We will use the following consequence of Geelen, Oxley, Vertigan, and Whittle [5, Theorem 8.4].

Theorem 3.1. *Let M and M' be $\text{GF}(4)$ -representable matroids with the properties that M is connected, M' is 3-connected, and M' is obtained from M by relaxing a circuit-hyperplane.*

- (i) *If M' has a $U_{2,4}$ -minor but no $\{U_{2,5}, U_{3,5}\}$ -minor, then M is binary.*
- (ii) *If M' has a $\{U_{2,5}, U_{3,5}\}$ -minor but no $U_{3,6}$ -minor, then M has no $\{U_{2,5}, U_{3,5}\}$ -minor.*

We can now prove the main result of this section.

Theorem 3.2. *Let M and M' be $\text{GF}(4)$ -representable matroids such that M is connected, M' is 3-connected, and M' is obtained from M by relaxing a circuit-hyperplane X . Then M' is either $U_{2,4}$ -fragile or $\{U_{2,5}, U_{3,5}\}$ -fragile. Moreover, M' has a basis X whose elements are nondeletable such that the elements of the cobasis $E(M') - X$ are noncontractible.*

Proof. First assume that M' has no $\{U_{2,5}, U_{3,5}\}$ -minor. By Lemma 2.5 and Theorem 3.1 (i), M' has a $U_{2,4}$ -minor and M has no $U_{2,4}$ -minor. Then it follows from Lemma 2.3 that M' is $U_{2,4}$ -fragile, and M' has a basis X whose elements are nondeletable such that the elements of the cobasis $E(M') - X$ are noncontractible.

We may now assume that M' has a $\{U_{2,5}, U_{3,5}\}$ -minor. Suppose that M also has a $\{U_{2,5}, U_{3,5}\}$ -minor, and assume that M is a minor-minimal matroid with respect to the hypotheses; that is, we assume that M has no proper minor M_0 such that M_0 is connected, M_0 has a $\{U_{2,5}, U_{3,5}\}$ -minor, and M_0 has a circuit-hyperplane whose relaxation M'_0 is 3-connected and $\text{GF}(4)$ -representable.

3.2.1. M is $\{U_{2,5}, U_{3,5}\}$ -fragile.

Proof of 3.2.1. Suppose that M has an element $e \in E(M) - X$ such that $M \setminus e$ has a $\{U_{2,5}, U_{3,5}\}$ -minor. If $M \setminus e$ is 3-connected, then we have a contradiction to the minimality of M . Therefore, by Lemma 2.11, $M \setminus e$ is 3-connected up to series and parallel pairs. Suppose that A is a non-trivial parallel class of $M \setminus e$. Suppose $A \subseteq X$. Then $A = X$ and $|A| = 2$ by Lemma 2.10, so we deduce that $M \setminus e$ is a parallel extension of $U_{2,5}$ and hence that $M' \setminus e$ has a $U_{2,6}$ -minor, a contradiction to the fact that the matroid M' obtained from M by relaxing X is $\text{GF}(4)$ -representable. Thus $A \subseteq E(M \setminus e) - X$ by Lemma 2.10. By duality, any non-trivial series class of $M \setminus e$ must be contained in X . Then, by Lemma 2.8, the matroid M_0 obtained from $M \setminus e$ by deleting all but one element of every non-trivial parallel class and contracting all

but one element of every non-trivial series class has a $\{U_{2,5}, U_{3,5}\}$ -minor. We deduce from Lemma 2.11 that M_0 is 3-connected. Then M_0 contradicts the minimality of M . Therefore $M \setminus e$ has no $\{U_{2,5}, U_{3,5}\}$ -minor for all $e \in E(M) - X$, and, by duality, M/e has no $\{U_{2,5}, U_{3,5}\}$ -minor for all $e \in X$, so M is $\{U_{2,5}, U_{3,5}\}$ -fragile. This completes the proof of 3.2.1.

Since M has a $\{U_{2,5}, U_{3,5}\}$ -minor, it follows from Theorem 3.1 (ii) that M' has a $U_{3,6}$ -minor, that is, $M'/C \setminus D \cong U_{3,6}$ for some subsets C and D . If $C \subseteq X$ and $D \subseteq E(M') - X$, then it follows from Lemma 2.2 that $U_{3,6}$ can be obtained from $M/C \setminus D$ by relaxing the circuit-hyperplane $X - C$. Hence $M/C \setminus D \cong P_6$, a contradiction because $M/C \setminus D$ is $\text{GF}(4)$ -representable but P_6 is not. Therefore $C \cap (E(M') - X)$ or $D \cap X$ is nonempty, so $M/C \setminus D = M'/C \setminus D \cong U_{3,6}$ by Lemma 2.2. This is a contradiction to 3.2.1 because any minor of M must also be $\{U_{2,5}, U_{3,5}\}$ -fragile, but for any e , both $U_{3,6} \setminus e$ and $U_{3,6}/e$ have a $\{U_{2,5}, U_{3,5}\}$ -minor. We conclude that M has no $\{U_{2,5}, U_{3,5}\}$ -minor. It now follows from Lemma 2.3 that M' is $\{U_{2,5}, U_{3,5}\}$ -fragile, and that M' has a basis X whose elements are nondeletable such that the elements of the cobasis $E(M') - X$ are noncontractible. \square

4. THE STRUCTURE OF $\{U_{2,5}, U_{3,5}\}$ -FRAGILE MATROIDS

4.1. Partial Fields and Constructions. We briefly state the necessary material on partial fields. For a more thorough treatment, we refer the reader to [13].

A *partial field* is a pair $\mathbb{P} = (R, G)$, where R is a commutative ring with unity, and G is a subgroup of the units of R with $-1 \in G$. A matrix with entries in G is a \mathbb{P} -matrix if $\det(D) \in G \cup \{0\}$ for any square submatrix D of A . We use $\langle X \rangle$ to denote the multiplicative subgroup of R generated by the subset X .

A rank- r matroid M on the ground set E is \mathbb{P} -representable if there is an $r \times |E|$ \mathbb{P} -matrix A such that, for each $r \times r$ submatrix D , the determinant of D is nonzero if and only if the corresponding subset of E is a basis of M . When this occurs, we write $M = M[A]$.

The 2-regular partial field is defined as follows.

$$\mathbb{U}_2 = (\mathbb{Q}(\alpha, \beta), \langle -1, \alpha, \beta, 1 - \alpha, 1 - \beta, \alpha - \beta \rangle),$$

where α, β are indeterminates.

It is well-known that any \mathbb{U}_2 -representable is $\text{GF}(4)$ -representable [11]. On the other hand, there are $\text{GF}(4)$ -representable that are not \mathbb{U}_2 -representable. We now define three such matroids. The matroid P_8 has a unique pair of disjoint circuit-hyperplanes; we let P_8^- denote the unique matroid obtained by relaxing one of these circuit-hyperplanes. We denote by F_7^- the matroid obtained from the non-Fano matroid F_7^- by relaxing a circuit-hyperplane. The $\text{GF}(4)$ -representable matroids $P_8^-, F_7^-, (F_7^-)^*$ are not \mathbb{U}_2 -representable. We

note that this can be deduced from [1] since $P_8^-, F_7^=, (F_7^=)^*$ are $\{U_{2,5}, U_{3,5}\}$ -fragile matroids. Since these matroids are not \mathbb{U}_2 -representable, we have the following lemma.

Lemma 4.1. *The class of \mathbb{U}_2 -representable matroids is contained in the class of $\text{GF}(4)$ -representable matroids with no $\{P_8^-, F_7^=, (F_7^=)^*\}$ -minor.*

To describe the structure of $\{U_{2,5}, U_{3,5}\}$ -fragile matroids as in [3], we need two constructions: the generalized Δ - Y exchange, and gluing on wheels. For a more thorough treatment of these constructions, we refer the reader to [11] and [2].

Loosely speaking, the operations of generalized Δ - Y exchange and gluing on wheels both involve gluing matroids together in along a common restriction. Let M_1 and M_2 be matroids with a common restriction A , where A is a modular flat of M_1 . The *generalized parallel connection* of M_1 and M_2 along A , denoted $P_A(M_1, M_2)$, is the matroid obtained by gluing M_1 and M_2 along A . It has ground set $E(M_1) \cup E(M_2)$, and a set F is a flat of $P_A(M_1, M_2)$ if and only if $F \cap E(M_i)$ is a flat of M_i for each i (see [10, Section 11.4]).

A subset S of $E(M)$ is a *segment* of M if every three-element subset of S is a triangle of M . Let M be a matroid with a k -element segment A . Intuitively, a generalized Δ - Y exchange on A turns the segment A into a k -element cosegment. To define the generalized Δ - Y exchange formally, we first recall the following definition of a family of matroids Θ_k from [11]. For $k \geq 3$, fix a basis $B = \{b_1, b_2, \dots, b_k\}$ of the rank- k projective geometry $PG(k-1, \mathbb{R})$, and choose a line L of $PG(k-1, \mathbb{R})$ that is freely placed relative to B . It follows from modularity that, for each i , the hyperplane spanned by $B - \{b_i\}$ meets L ; we let a_i be the point of intersection. Let $A = \{a_1, a_2, \dots, a_k\}$, and let Θ_k be the matroid obtained by restricting $PG(k-1, \mathbb{R})$ to the set $A \cup B$. Note that the matroid Θ_k has A as a modular k -point segment A , so the generalized parallel connection of Θ_k and M along A is well-defined. If the k -element segment A is coindependent in M , then we define the matroid $\Delta_A(M)$ to be the matroid obtained from $P_A(\Theta_k, M) \setminus A$ by relabeling the elements of $E(\Theta_k) - A$ by A in the natural way, and we say that $\Delta_A(M)$ is obtained from M by performing a *generalized Δ - Y exchange* on A . For a matroid M with an independent cosegment A , a *generalized Y - Δ exchange on A* , denoted by $\nabla_A(M)$, is defined to be the matroid $(\Delta_A(M^*))^*$.

We use the following results on representability and the minor operations.

Lemma 4.2. [11, Lemma 3.7] *Let \mathbb{P} be a partial field. Then M is \mathbb{P} -representable if and only if $\Delta_A(M)$ is \mathbb{P} -representable.*

Lemma 4.3. [11, Lemma 2.13] *Suppose that $\Delta_A(M)$ is defined. If $x \in A$ and $|A| \geq 3$, then $\Delta_{A-x}(M \setminus x)$ is also defined, and $\Delta_A(M)/x = \Delta_{A-x}(M \setminus x)$.*

Lemma 4.4. [11, Lemma 2.16] *Suppose that $\Delta_A(M)$ is defined.*

- (i) If $x \in E(M) - A$ and A is coindependent in $M \setminus x$, then $\Delta_A(M \setminus x)$ is defined and $\Delta_A(M) \setminus x = \Delta_A(M \setminus x)$.
- (ii) If $x \in E(M) - \text{cl}(A)$, then $\Delta_A(M/x)$ is defined and $\Delta_A(M)/x = \Delta_A(M/x)$.

Lemma 4.5. [11, Lemma 2.15] Suppose that $x \in \text{cl}(A) - A$ and let a be an arbitrary element of the k -element segment A . Then $\Delta_A(M)/x$ equals the 2-sum, with basepoint p , of a copy of $U_{k-1,k+1}$ with groundset $A \cup p$ and the matroid obtained from $M/x \setminus (A - a)$ by relabeling a as p .

The next result implies that every $\{U_{2,5}, U_{3,5}\}$ -fragile matroid is 3-connected up to series and parallel classes.

Lemma 4.6. [9, Proposition 4.3] Let M be a matroid with a 2-separation (A, B) , and let N be a 3-connected minor of M . Assume $|E(N) \cap A| \geq |E(N) \cap B|$. Then $|E(N) \cap B| \leq 1$. Moreover, unless B is a parallel or series class, there is an element $x \in B$ such that both $M \setminus x$ and M/x have a minor isomorphic to N .

The following is an easy consequence of the property that strictly $\{U_{2,5}, U_{3,5}\}$ -fragile matroids are 3-connected up to parallel and series classes.

Lemma 4.7. Let M be a strictly $\{U_{2,5}, U_{3,5}\}$ -fragile matroid with at least 8 elements. If S is a triangle or 4-element segment of M such that $E(M) - S$ is not a series or parallel class of M , then S is coindependent in M . If C is a triad or 4-element cosegment of M such that $E(M) - S$ is not a series or parallel class of M , then C is independent.

Let M be a $\{U_{2,5}, U_{3,5}\}$ -fragile matroid. A segment S of M is *allowable* if S is coindependent and some element of S is nondeletable. A cosegment C of M is *allowable* if the segment C of M^* is allowable. In [3], it was shown that we can obtain a new $\{U_{2,5}, U_{3,5}\}$ -fragile \mathbb{U}_2 -representable matroid from an old $\{U_{2,5}, U_{3,5}\}$ -fragile \mathbb{U}_2 -representable matroid by performing a generalized Δ - Y exchange on an allowable segment. We will prove an analogous result for $\{U_{2,5}, U_{3,5}\}$ -fragile $\text{GF}(4)$ -representable matroids with no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor.

Let \mathcal{U} be the class of $\text{GF}(4)$ -representable matroids with no $\{U_{2,5}, U_{3,5}\}$ -minor. The class of *sixth-root-of-unity* matroids is the class of matroids that are representable over both $\text{GF}(3)$ and $\text{GF}(4)$. Semple and Whittle [14, Theorem 5.2] showed that \mathcal{U} is the class of matroids that can be obtained by taking direct sums and 2-sums of binary and sixth-root-of-unity matroids.

Lemma 4.8. Let M be a matroid in the class \mathcal{U} . If M' is obtained from M by performing a generalized Δ - Y exchange or a generalized Y - Δ exchange, then $M' \in \mathcal{U}$.

Proof. Suppose that there exists a matroid $M \in \mathcal{U}$ with a coindependent segment A such that $\Delta_A(M) \notin \mathcal{U}$. Among all counterexamples, suppose that M has been chosen so that $|E(M)|$ is as small as possible. Then M

is not 3-connected, since any 3-connected member of \mathcal{U} is either a binary or sixth-root-of-unity matroid. Hence $\Delta_A(M)$ is a binary or sixth-root-of-unity matroid by Lemma 4.2, so $\Delta_A(M) \in \mathcal{U}$, contradicting the assumption that M is a counterexample. Therefore M is not 3-connected. Now either $M = M_1 \oplus M_2$ or $M = M_1 \oplus_2 M_2$ for some $M_1, M_2 \in \mathcal{U}$ with $|E(M_i)| < |E(M)|$ for each $i \in \{1, 2\}$. Moreover, we may assume that M_1 and M_2 have been chosen so that the segment A of M is contained in $E(M_1)$. Now either $\Delta_A(M) = \Delta_A(M_1) \oplus M_2$ or $\Delta_A(M) = \Delta_A(M_1) \oplus_2 M_2$. Since $|E(M_1)| < |E(M)|$, it follows that $\Delta_A(M_1) \in \mathcal{U}$. Hence $\Delta_A(M) \in \mathcal{U}$. Since \mathcal{U} is closed under duality, the result follows. \square

Lemma 4.9. *Let M be a $\{U_{2,5}, U_{3,5}\}$ -fragile GF(4)-representable matroid with no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor. If A is an allowable segment of M with $|A| \in \{3, 4\}$, then $\Delta_A(M)$ is a $\{U_{2,5}, U_{3,5}\}$ -fragile GF(4)-representable matroid with no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor. Moreover, A is an allowable cosegment of $\Delta_A(M)$.*

Proof. The proof that $\Delta_A(M)$ is a $\{U_{2,5}, U_{3,5}\}$ -fragile GF(4)-representable matroid where A is an allowable cosegment of $\Delta_A(M)$ closely follows the proof of [3, Lemma 2.21]. The only difference is where the proof of [3, Lemma 2.21] uses the fact that a \mathbb{U}_2 -representable matroid with no $\{U_{2,5}, U_{3,5}\}$ -minor is near-regular and the class of near-regular matroids is closed under the generalized Δ -Y exchange, we instead use Lemma 4.8.

We must also show that $\Delta_A(M)$ has no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor. This follows for $|E(M)| \leq 9$ from the generation of the 3-connected $\{U_{2,5}, U_{3,5}\}$ -fragile GF(4)-representable matroids with no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor on at most 9 elements (see the Appendix), since all such matroids are \mathbb{U}_2 -representable. Suppose that M is a minimum-sized counterexample, so $\Delta_A(M)$ has a $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor and $\Delta_A(M)$ has at least ten elements. Then $\Delta_A(M)$ has a minor N , obtained by deleting or contracting an element x say, that also has a $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor. Since $\Delta_A(M)$ is $\{U_{2,5}, U_{3,5}\}$ -fragile it follows that the minor N is also $\{U_{2,5}, U_{3,5}\}$ -fragile. Suppose that $N = \Delta_A(M)/x$. Suppose that $x \in A$. Then $\Delta_A(M)/x = \Delta_{A-x}(M \setminus x)$ by Lemma 4.3, a contradiction since M is a minimum-sized counterexample. Next suppose that $x \in \text{cl}(A) - A$. Since N is $\{U_{2,5}, U_{3,5}\}$ -fragile it follows from Lemma 4.5 and Proposition 4.6 that $|A| = 4$ and $M/x \setminus (A - a) \cong U_{1,n}$ for some $n \geq 2$. Hence $\Delta_A(M)$ has no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor, a contradiction. We may now assume $x \in E(M) - \text{cl}(A)$. Then $\Delta_A(M)/x = \Delta_A(M/x)$ by Lemma 4.4, a contradiction since M is a minimum-sized counterexample. We deduce that $N = \Delta_A(M) \setminus x$, and we may assume that any minor obtained from $\Delta_A(M)$ by contracting an element has no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor. Now if $x \in A$, then $A - x$ is a series class of $\Delta_A(M) \setminus x$, so there is some $y \in A$ such that $\Delta_A(M)/y$ has a $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor, a contradiction. Therefore $x \notin A$. If A is not conindependent in $M \setminus x$, then it follows from Lemma 4.7 that $\Delta_A(M)$ has no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor, a contradiction. Therefore A is conindependent in $M \setminus x$, so $\Delta_A(M) \setminus x = \Delta_A(M \setminus x)$

by Lemma 4.4, a contradiction since M is a minimum-sized counterexample. \square

Let M be a matroid, and (a, b, c) an ordered subset of $E(M)$ such that $T = \{a, b, c\}$ is a triangle. Let $r \geq 3$ be a positive integer, and, when $r = 3$, we fix a vertex of \mathcal{W}_3 to be the center, so we can refer to rim and spoke elements of $M(\mathcal{W}_3)$. Let N be obtained from $M(\mathcal{W}_r)$ by relabeling some triangle as $\{a, b, c\}$, where a, c are spoke elements, and let $X \subseteq \{a, b, c\}$ such that $b \in X$. We say the matroid $M' := P_T(M, N) \setminus X$ is obtained from M by *gluing an r -wheel onto (a, b, c)* . We also say that M^* is obtained from N^* by gluing a wheel onto the triad T . Suppose that T_1, T_2, \dots, T_n are ordered triangles of M . We say M' can be obtained from M by *gluing wheels onto T_1, T_2, \dots, T_n* if, for some subset J of $\{1, 2, \dots, n\}$, M' can be obtained from M by a sequence of moves, where each move consists of gluing an r_j -wheel onto T_j for $j \in J$. Note that the spoke elements of a triangle in this sequence are only possibly deleted when they do not appear in any subsequent triangle in the sequence.

Lemma 4.10. *Let M be a $\{U_{2,5}, U_{3,5}\}$ -fragile GF(4)-representable matroid with no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor. Let $A = \{a, b, c\}$ be an allowable triangle of M , where b is nondeletable. If M' is obtained from M by gluing an r -wheel onto (a, b, c) , where $X \subseteq \{a, b, c\}$ is such that $b \in X$, then M' is a $\{U_{2,5}, U_{3,5}\}$ -fragile GF(4)-representable matroid with no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor. Moreover, $F = E(\mathcal{W}_r) - X$ is the set of elements of a fan, the spoke elements of F are noncontractible in M' , and the rim elements of F are nondeletable in M' .*

Proof. The proof is the same as [3, Lemma 2.22] except that we use Lemma 4.9 instead of [3, Lemma 2.21]. \square

4.2. Path sequences. We can now describe a family of $\{U_{2,5}, U_{3,5}\}$ -fragile GF(4)-representable matroids with no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor obtained by performing generalized Δ - Y exchanges and gluing on wheels. In fact, the matroids in this family are \mathbb{U}_2 -representable and were first described in [3]. Each matroid in this family has an associated path of 3-separations that we need to describe in order to define the family.

We call the set $X \subseteq E(M)$ *fully closed* if X is closed in both M^* and M . The *full closure* of X , denoted $\text{fcl}_M(X)$, is the intersection of all fully closed sets containing X . The full closure of X can be obtained from X by repeatedly taking closure and coclosure until no new elements are added. We call X a *path-generating* set if X is a 3-separating set of M such that $\text{fcl}_M(X) = E(M)$. A path-generating set X thus gives rise to a natural path of 3-separating sets (P_1, \dots, P_m) , where $P_1 = X$ and each step P_i is either the closure or coclosure of the 3-separating set $P_1 \cup \dots \cup P_{i-1}$.

Let X be an allowable cosegment of M . A matroid Q is an *allowable series extension of M along X* if $M = Q/Z$ and, for every element z of Z , there is some element x of X such that x is \mathcal{N} -contractible in M and z is in

series with x in Q . We also say that Q^* is an *allowable parallel extension of M^* along X* .

Let N be a matroid with a path-generating allowable segment or cosegment A . We say that M is obtained from N by a Δ - ∇ -step along A if, up to duality, M is obtained from N by performing a non-empty allowable parallel extension along A , followed by a generalized Δ - Y exchange on A .

Let X_8 be the matroid obtained from $U_{2,5}$ by choosing a 4-element segment C , adding a point in parallel with each of three distinct points of C , then performing a generalized Δ - Y -exchange on C . In what follows, S will be the elements of the 4-element segment of X_8 , and C the elements of the 4-element cosegment of X_8 , so $E(X_8) = S \cup C$. We will build matroids from X_8 by performing a sequence of Δ - ∇ -steps along $A \in \{S, C\}$. Note that, in such matroids, each of S and C can be either a segment or a cosegment.

A sequence of matroids M_1, \dots, M_n is called a *path sequence* if the following conditions hold:

- (P1) $M_1 = X_8$; and
- (P2) For each $i \in \{1, \dots, n-1\}$, there is some 4-element path-generating segment or cosegment $A \in \{S, C\}$ of M_i such that either:
 - (a) M_{i+1} is obtained from M_i by a Δ - ∇ -step along A ; or
 - (b) M_{i+1} is obtained from M_i by gluing a wheel onto an allowable subset A' of A .

Note in (P2) that each Δ - ∇ -step described in (a) increases the number of elements by at least one, and that the wheels in (b) are only glued onto allowable subsets of 4-element segments or cosegments.

We say that a path sequence M_1, \dots, M_n *describes* a matroid M if $M_n \cong M$. We also say that M is a matroid *described by* a path sequence if there is some path sequence that describes M . Let \mathcal{P} denote the class of matroids such that $M \in \mathcal{P}$ if and only if there is some path sequence M_1, \dots, M_n that describes a matroid M' such that M can be obtained from M' by some, possibly empty, sequence of allowable parallel and series extensions. Since X_8 is self-dual, it is easy to see that the sequence of dual matroids M_1^*, \dots, M_n^* of a path sequence M_1, \dots, M_n is also a path sequence. Thus the class \mathcal{P} is closed under duality.

We denote by Y_8 the unique matroid obtained from X_8 by performing a Y - Δ -exchange on an allowable triad. We will prove the following result.

Theorem 4.11. *If M is a 3-connected $\{U_{2,5}, U_{3,5}\}$ -fragile GF(4)-representable matroid that has an $\{X_8, Y_8, Y_8^*\}$ -minor but no $\{P_8^-, F_7^=, (F_7^=)^*\}$ -minor, then there is some path sequence that describes M .*

The proof of Theorem 4.11 closely follows the proof of [3, Corollary 4.3]. The strategy is to show that a minor-minimal counterexample has at most 12 elements. Let M be a GF(4)-representable $\{U_{2,5}, U_{3,5}\}$ -fragile matroid M with an $\{X_8, Y_8, Y_8^*\}$ -minor but no $\{P_8^-, F_7^=, (F_7^=)^*\}$ -minor. Suppose that M is a minimum-sized matroid that is not in the class \mathcal{P} . Then M is

3-connected because \mathcal{P} is closed under series and parallel extensions. Moreover, the dual M^* is also not in \mathcal{P} because \mathcal{P} is closed under duality. Thus, by the Splitter Theorem and duality, we may assume there is some element x of M such that $M \setminus x$ is also a 3-connected $\text{GF}(4)$ -representable $\{U_{2,5}, U_{3,5}\}$ -fragile matroid with an $\{X_8, Y_8, Y_8^*\}$ -minor but no $\{P_8^-, F_7^=, (F_7^=)^*\}$ -minor. By the assumption that M is minimum-sized with respect to being outside the class \mathcal{P} , it follows that $M \setminus x \in \mathcal{P}$. Thus $M \setminus x$ is described by a path sequence M_1, \dots, M_n . The next lemma [3, Lemma 6.3] identifies the three possibilities for the position of x in M relative to the path of 3-separations associated with M_1, \dots, M_n .

Lemma 4.12. *Let M and $M \setminus x$ be 3-connected $\{U_{2,5}, U_{3,5}\}$ -fragile matroids. If $M \setminus x$ is described by a path sequence with associated path of 3-separations \mathbf{P} , then either:*

- (i) *there is some 3-separation (X, Y) displayed by \mathbf{P} such that $x \in \text{cl}(X)$ and $x \in \text{cl}(Y)$; or*
- (ii) *there is some 3-separation (X, Y) displayed by \mathbf{P} such that $x \notin \text{cl}(X)$ and $x \notin \text{cl}(Y)$; or*
- (iii) *for each 3-separation (R, G) of M displayed by \mathbf{P} , there is some $X \in \{R, G\}$ such that $x \in \text{cl}_M(X)$ and $x \in \text{cl}_M^*(X)$.*

The proofs of the next three lemmas follow the proofs of [3, Lemma 7.4], [3, Lemma 8.7], and [3, Lemma 9.7] but use Lemma 4.9 above instead of [3, Lemma 2.21].

Lemma 4.13. *Lemma 4.12 (i) does not hold.*

Lemma 4.14. *If Lemma 4.12 (ii) holds, then $|E(M \setminus x)| \leq 10$.*

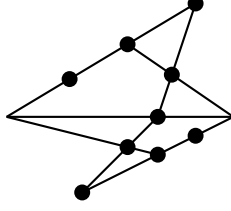
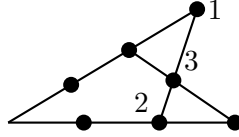
Lemma 4.15. *If Lemma 4.12 (iii) holds, then $|E(M \setminus x)| \leq 11$.*

Proof of Theorem 4.11. In view of the last three lemmas, it suffices to verify that \mathcal{P} contains each 3-connected $\{U_{2,5}, U_{3,5}\}$ -fragile $\text{GF}(4)$ -representable matroid with an $\{X_8, Y_8, Y_8^*\}$ -minor and no $\{P_8^-, F_7^=, (F_7^=)^*\}$ -minor having at most 12 elements. This is done in the Appendix. \square

4.3. Fan extensions. The following theorem describes the structure of the matroids with no $\{X_8, Y_8, Y_8^*\}$ -minor. Note that $M_{9,9}$ is the rank-4 matroid on 9 elements in Figure 1. The matroid $M_{7,1}$ is the 7-element matroid that is obtained from Y_8 by deleting the unique point that is contained in the two 4-element segments of Y_8 . We label the points of a triangle of $M_{7,1}$ by $\{1, 2, 3\}$ as in Figure 2.

Theorem 4.16. *Let M' be a 3-connected $\{U_{2,5}, U_{3,5}\}$ -fragile $\text{GF}(4)$ -representable matroid with no $\{P_8^-, F_7^=, (F_7^=)^*\}$ -minor. Then M' is isomorphic to a matroid M for which at least one of the following holds:*

- (i) *M has an $\{X_8, Y_8, Y_8^*\}$ -minor;*
- (ii) *$M \in \{M_{9,9}, M_{9,9}^*\}$;*

FIGURE 1. The matroid $M_{9,9}$.FIGURE 2. The matroid $M_{7,1}$.

- (iii) M or M^* can be obtained from $U_{2,5}$ (with ground set $\{a, b, c, d, e\}$) by gluing wheels to $(a, c, b), (a, d, b), (a, e, b)$;
- (iv) M or M^* can be obtained from $U_{2,5}$ (with ground set $\{a, b, c, d, e\}$) by gluing wheels to $(a, b, c), (c, d, e)$;
- (v) M or M^* can be obtained from $M_{7,1}$ by gluing a wheel to $(1, 3, 2)$.

Proof. Assume M has no $\{X_8, Y_8, Y_8^*\}$ -minor. For (ii), we show in Lemma 1 of the Appendix that the matroids $M_{9,9}$ and $M_{9,9}^*$ are splitters for the class of 3-connected $\{U_{2,5}, U_{3,5}\}$ -fragile $\text{GF}(4)$ -representable matroids with no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor.

We may therefore assume M has no $\{M_{9,9}, M_{9,9}^*, X_8, Y_8, Y_8^*\}$ -minor. To show that (iii), (iv), or (v) holds, we use the main result of [2] called the “Fan Lemma”, which reduces the proof to showing that extensions and coextensions of the 9-elements matroids with this structure also have this structure. These verifications are completed in Lemmas 2 through 7 of the Appendix. \square

5. FROM FRAGILITY TO RELAXATIONS

We use the following result of Mayhew, Whittle, and Van Zwam [9, Lemma 8.2].

Lemma 5.1. *Let M be a 3-connected $U_{2,4}$ -fragile matroid that has no $\{U_{2,6}, U_{4,6}\}$ -minor. Then exactly one of the following holds.*

- (i) M has rank or corank two;
- (ii) M has an $\{F_7^-, (F_7^-)^*\}$ -minor;
- (iii) M has rank and corank at least 3 and is a whirl.

We show next that $P_8^-, F_7^-, (F_7^-)^*$ do not arise from circuit-hyperplane relaxation of a $\text{GF}(4)$ -representable matroid.

Lemma 5.2. *Let M and M' be 3-connected $\text{GF}(4)$ -representable matroids with the property that M' is obtained from M by relaxing a circuit-hyperplane X . Then M' has no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor.*

Proof. Assume that M' has a $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor. Since M' is obtained from M by relaxing X , it follows from Theorem 3.2 and Lemma 5.1 that M' is $\{U_{2,5}, U_{3,5}\}$ -fragile. Each of the matroids in $\{P_8^-, F_7^-, (F_7^-)^*\}$ has a $\{U_{2,5}, U_{3,5}\}$ -minor, so if $M'/C \setminus D \cong N'$ for some $N' \in \{P_8^-, F_7^-, (F_7^-)^*\}$, then $C \subseteq X$ and $D \subseteq E(M') - X$ since the elements of X are nondeletable and the elements of $E(M') - X$ are noncontractible by Theorem 3.2. But then it follows from Lemma 2.2 that N' can be obtained from $M/C \setminus D$ by relaxing the circuit-hyperplane $X - C$. It follows that $M/C \setminus D \cong N$ for some $N \in \{P_8^-, F_7^-, (F_7^-)^*\}$, a contradiction because M is $\text{GF}(4)$ -representable. \square

We can now describe the structure of the $\text{GF}(4)$ -representable matroids that are circuit-hyperplane relaxations of $\text{GF}(4)$ -representable matroids.

Theorem 5.3. *Let M and M' be $\text{GF}(4)$ -representable matroids such that M is connected, M' is 3-connected, and M' is obtained from M by relaxing a circuit-hyperplane. Then at least one of the following holds.*

- (a) M' is a whirl;
- (b) $M' \in \{M_{9,9}, M_{9,9}^*\}$;
- (c) M' or $(M')^*$ can be obtained from $U_{2,5}$ (with groundset $\{a, b, c, d, e\}$) by gluing wheels to $(a, c, b), (a, d, b)$;
- (d) M' or $(M')^*$ can be obtained from $U_{2,5}$ (with groundset $\{a, b, c, d, e\}$) by gluing wheels to $(a, b, c), (c, d, e)$;
- (e) M' or $(M')^*$ can be obtained from $M_{7,1}$ by gluing a wheel to $(1, 3, 2)$;
- (f) there is some path sequence that describes M' .

Proof. It follows from Theorem 3.2 that M' is either $U_{2,4}$ -fragile or $\{U_{2,5}, U_{3,5}\}$ -fragile. If M' is $U_{2,4}$ -fragile, then it follows from Lemma 5.1 that M' is a whirl. We may therefore assume that M' is $\{U_{2,5}, U_{3,5}\}$ -fragile. It follows from Lemma 5.2 that M' has no $\{P_8^-, F_7^-, (F_7^-)^*\}$ -minor. Then, by Theorem 4.16 and Theorem 3.2, one of (b) through (e) holds or else M' has an $\{X_8, Y_8, Y_8^*\}$ -minor. Note that outcome (iii) of Theorem 4.16 corresponds to outcome (c) here, since a matroid or its dual that is obtained from $U_{2,5}$ (with groundset $\{a, b, c, d, e\}$) by gluing wheels onto all three of the triangles $(a, c, b), (a, d, b), (a, e, b)$ does not have a basis of nondeletable elements and a cobasis $E(M') - H$ of noncontractible elements, and therefore cannot be the obtained by relaxing a circuit-hyperplane. We can see this by the following counting argument. Observe that the rank of a matroid obtained from $U_{2,5}$ (with groundset $\{a, b, c, d, e\}$) by gluing wheels A, B and C onto the triangles $(a, c, b), (a, d, b), (a, e, b)$ is $r(A) + r(B) + r(C) - 4$. But the nondeletable elements of this matroid are precisely the rim elements of the wheels of which there are $r(A) + r(B) + r(C) - 3$. Hence any cobasis must

contain a nondeletable element. Finally, if M' has an $\{X_8, Y_8, Y_8^*\}$ -minor, then it follows from Theorem 4.11 that (f) holds. \square

We can now show that if M and M' are $\text{GF}(4)$ -representable matroids such that M' is obtained from M by relaxing a circuit-hyperplane, then M' has path width 3.

Proof of Theorem 1.1. If M is not connected, then it follows from Lemma 2.9 that M' has path width 3. We may therefore assume that M is connected. Then, by Lemma 2.11, M' can be obtained from a matroid in Theorem 5.3 (a) - (f) by performing some, possibly empty, sequence of series or parallel extensions. The result now follows from the fact that all the matroids in Theorem 5.3 (a) - (f) have path width 3. \square

6. FORBIDDEN SUBMATRICES

In this section, we will prove our second characterization, Theorem 1.2. Let M be a $\text{GF}(4)$ -representable matroid with a circuit-hyperplane X . Choose $e \in X$ and $f \in E - X$ such that $B = (X - e) \cup f$ is a basis of M . Then we can find a reduced $\text{GF}(4)$ -representation of M in block form,

$$C = \begin{matrix} & (E-X)-f & e \\ \begin{matrix} X-e \\ f \end{matrix} & \begin{bmatrix} A & \mathbf{1} \\ \mathbf{1}^T & 0 \end{bmatrix} \end{matrix}.$$

Here A is an $(X - e) \times ((E - X) - f)$ matrix over $\text{GF}(4)$, and we have scaled so that every non-zero entry in the row labelled by f and the column labelled by e is 1. We denote by A_{ij} the entry in row i and column j of A .

Let M' be the matroid obtained from M by relaxing the circuit-hyperplane X . If M' is $\text{GF}(4)$ -representable, then we can find a reduced representation of M' in block form,

$$C' = \begin{matrix} & (E-X)-f & e \\ \begin{matrix} X-e \\ f \end{matrix} & \begin{bmatrix} A' & \mathbf{1} \\ \mathbf{1}^T & \omega \end{bmatrix} \end{matrix}.$$

We have scaled the rows and columns of the matrix such that the entry in the row labelled by f and column labelled by e is $\omega \in \text{GF}(4) - \{0, 1\}$, and every remaining entry in row e and column f is a 1.

We omit the straightforward proof of the following lemma.

Lemma 6.1. $A_{ij} = 0$ if and only if $A'_{ij} = 0$.

Next we show that the only non-zero entries of A' are 1 and ω .

Lemma 6.2. $A'_{ij} \neq \omega + 1$.

Proof. Suppose $A'_{ij} = \omega + 1$. Then C' has a submatrix

$$C'[\{i, f\}, \{e, j\}] = \begin{matrix} & j & e \\ \begin{matrix} i \\ f \end{matrix} & \begin{bmatrix} \omega + 1 & 1 \\ 1 & \omega \end{bmatrix} \end{matrix},$$

which has determinant zero. Therefore $B \triangle \{e, f, i, j\}$ is not a basis of the matroid $M[I|C']$. But the corresponding submatrix of C is

$$C[\{i, f\}, \{e, j\}] = \begin{matrix} & j & e \\ \begin{matrix} i \\ f \end{matrix} & \begin{bmatrix} x & 1 \\ 1 & 0 \end{bmatrix} \end{matrix},$$

for some non-zero x . Since $C[\{i, f\}, \{e, j\}]$ has non-zero determinant, $B \triangle \{e, f, i, j\}$ is a basis of M , and hence of M' . Therefore $M' \neq M[I|C']$. \square

Lemma 6.3. $A_{ij} = A_{ik}$ if and only if $A'_{ij} = A'_{ik}$. Similarly, $A_{ij} = A_{kj}$ if and only if $A'_{ij} = A'_{kj}$.

Proof. We show that $A_{ij} = A_{ik}$ implies that $A'_{ij} = A'_{ik}$. The proof of the converse, and the proof of the second statement proceed by similar easy arguments. Suppose that $A_{ij} = A_{ik}$. Then C has a submatrix

$$C[\{i, f\}, \{j, k\}] = \begin{matrix} & j & e \\ \begin{matrix} i \\ f \end{matrix} & \begin{bmatrix} x & x \\ 1 & 1 \end{bmatrix} \end{matrix},$$

for some non-zero x . Since $C[\{i, f\}, \{j, k\}]$ has zero determinant, $B \triangle \{f, i, j, k\}$ is not a basis of M , and hence not a basis of $M' = M[I|C']$. Therefore $\det(C'[\{i, f\}, \{j, k\}]) = 0$, so it follows that $A'_{ij} = A'_{ik}$. \square

The following lemma on diagonal submatrices will be used frequently.

Lemma 6.4. *Let*

$$\begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix} \text{ and } \begin{bmatrix} x & 0 \\ 0 & y \end{bmatrix}$$

be corresponding submatrices of A and A' , where a, b, x, y are non-zero entries. Then $a = b$ if and only if $x \neq y$.

Proof. Adjoining e and f to the specified 2×2 submatrices, we get the 3×3 submatrices

$$\begin{bmatrix} a & 0 & 1 \\ 0 & b & 1 \\ 1 & 1 & 0 \end{bmatrix} \text{ and } \begin{bmatrix} x & 0 & 1 \\ 0 & y & 1 \\ 1 & 1 & \omega \end{bmatrix}.$$

These matrices have determinants $a + b$ and $xy\omega + x + y$. Thus if $a = b$, then $x \neq y$. Conversely, if $x \neq y$, then $\{x, y\} = \{1, \omega\}$ so $xy\omega + x + y = \omega^2 + \omega + 1 = 0$. Hence $a = b$. \square

We can now identify all of the forbidden submatrices. We use Lemma 6.3 to identify the first such matrix in the following lemma.

Lemma 6.5. *Neither A nor A^T has a submatrix of the form*

$$\begin{bmatrix} x & y & z \end{bmatrix},$$

where x, y, z are distinct non-zero entries.

Proof. By Lemma 6.3, the corresponding submatrix of A' must also have the form

$$\begin{bmatrix} x & y & z \end{bmatrix},$$

where x, y, z are distinct non-zero entries, which is a contradiction to Lemma 6.2. \square

We now use Lemma 6.3 and Lemma 6.4 to find several more forbidden submatrices.

Lemma 6.6. *A has no submatrices of the following forms, where x, y , and z are distinct non-zero entries.*

$$\begin{aligned} & (i) \begin{bmatrix} x & x & 0 \\ x & 0 & x \end{bmatrix}; (ii) \begin{bmatrix} x & x & 0 \\ x & 0 & y \end{bmatrix}; (iii) \begin{bmatrix} x & x & 0 \\ y & 0 & y \end{bmatrix}; (iv) \begin{bmatrix} x & y & 0 \\ x & 0 & y \end{bmatrix}; \\ & (v) \begin{bmatrix} x & 0 & 0 \\ 0 & y & z \end{bmatrix}; (vi) \begin{bmatrix} x & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & x \end{bmatrix}; (vii) \begin{bmatrix} x & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & y \end{bmatrix}; (viii) \begin{bmatrix} x & 0 & 0 \\ 0 & y & 0 \\ 0 & 0 & z \end{bmatrix}. \end{aligned}$$

Proof. Suppose A has the submatrix (i). By applying Lemma 6.3 to the rows and the first column, we deduce that the corresponding submatrix of A' has the form

$$\begin{bmatrix} y & y & 0 \\ y & 0 & y \end{bmatrix},$$

where y is a non-zero entry, a contradiction of Lemma 6.4.

Suppose A has the submatrix (ii). By applying Lemma 6.3 to the rows and the first column, and since A' has at most two distinct non-zero entries by Lemma 6.2, we deduce that the corresponding submatrix of A' has the form

$$\begin{bmatrix} a & a & 0 \\ a & 0 & b \end{bmatrix},$$

where a and b are the two non-zero entries of A' , a contradiction to Lemma 6.4.

The proofs for (iii) and (iv) are similar to that for (ii). We omit the details.

Suppose A has the submatrix (v). Then, by two applications of Lemma 6.4, the corresponding submatrix of A' must have the form

$$\begin{bmatrix} a & 0 & 0 \\ 0 & a & a \end{bmatrix},$$

for some non-zero entry a . This is a contradiction to Lemma 6.3.

Suppose A has the submatrix (vi). By Lemma 6.4, the corresponding submatrix of A' must be a diagonal matrix with distinct non-zero entries, a contradiction to Lemma 6.2.

Suppose A has the submatrix (vii). Applying Lemma 6.4 to the two submatrices the form

$$\begin{bmatrix} x & 0 \\ 0 & y \end{bmatrix},$$

it follows that the corresponding submatrix of A' is

$$\begin{bmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & a \end{bmatrix},$$

for some a , which is a contradiction to Lemma 6.4.

Suppose A has the submatrix (viii). Then the corresponding submatrix of A' is

$$\begin{bmatrix} a & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & a \end{bmatrix},$$

for some a . Adjoining e and f , we have a submatrix of C ,

$$\begin{bmatrix} x & 0 & 0 & 1 \\ 0 & y & 0 & 1 \\ 0 & 0 & z & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix},$$

which has zero determinant, while the corresponding submatrix of C' ,

$$\begin{bmatrix} a & 0 & 0 & 1 \\ 0 & a & 0 & 1 \\ 0 & 0 & a & 1 \\ 1 & 1 & 1 & \omega \end{bmatrix},$$

has non-zero determinant, a contradiction. \square

Lemma 6.7. *A has no submatrices of the following forms, where x , y , and z are distinct non-zero entries.*

$$(i) \begin{bmatrix} x & y \\ 0 & x \end{bmatrix}; (ii) \begin{bmatrix} x & y \\ y & x \end{bmatrix}; (iii) \begin{bmatrix} x & x \\ y & z \end{bmatrix}; (iv) \begin{bmatrix} x & y \\ z & x \end{bmatrix}.$$

Proof. Suppose A has the submatrix (i). Then, adjoining e and f , we see that C has the following submatrix with non-zero determinant.

$$\begin{bmatrix} x & y & 1 \\ 0 & x & 1 \\ 1 & 1 & 0 \end{bmatrix}.$$

But then, by Lemma 6.3, the corresponding submatrix of C' must have the following form.

$$\begin{bmatrix} a & b & 1 \\ 0 & a & 1 \\ 1 & 1 & \omega \end{bmatrix},$$

where a and b are distinct non-zero entries. This gives a contradiction because this submatrix of C' has zero determinant. A similar proof handles (ii).

Suppose A has the submatrix (iii). Then, by Lemma 6.3, in the corresponding submatrix of A' , the entries in the first row are the same and the entries in the second row are different. But, by Lemma 6.2, there are only two distinct non-zero entries in A' , so the entries are the same in one of the columns of A' , which is a contradiction to Lemma 6.3.

Suppose A has the submatrix (iv). Note that this submatrix has zero determinant. By Lemma 6.3, the corresponding submatrix of A' must have the following form.

$$\begin{bmatrix} a & b \\ b & a \end{bmatrix},$$

where a and b are distinct non-zero entries. But this submatrix of A' has non-zero determinant, a contradiction. \square

Finally, we find two more 3×3 forbidden submatrices of A .

Lemma 6.8. *A has no submatrices of the following forms, where x , y , and z are distinct non-zero entries.*

$$(i) \begin{bmatrix} x & y & x \\ y & y & 0 \\ x & 0 & 0 \end{bmatrix}; (ii) \begin{bmatrix} x & y & x \\ y & y & 0 \\ x & 0 & z \end{bmatrix}.$$

Proof. Suppose that A has the submatrix (i). Then, adjoining e and f , we see that C has the submatrix

$$\begin{bmatrix} x & y & x & 1 \\ y & y & 0 & 1 \\ x & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix},$$

which has zero determinant. The corresponding submatrix of C' is

$$\begin{bmatrix} a & b & a & 1 \\ b & b & 0 & 1 \\ a & 0 & 0 & 1 \\ 1 & 1 & 1 & \omega \end{bmatrix},$$

for distinct $a, b \in \{1, \omega\}$. This submatrix of C has non-zero determinant, a contradiction.

Suppose that A has the submatrix (ii). Note that the determinant of this submatrix is not zero. By Lemma 6.2 and Lemma 6.3, the corresponding submatrix of A' is

$$\begin{bmatrix} a & b & a \\ b & b & 0 \\ a & 0 & b \end{bmatrix},$$

for distinct $a, b \in \{1, \omega\}$. This submatrix of A' has zero determinant, which is a contradiction. \square

To prove the main theorem of this section, we need the following theorem [4, Theorem 5.1].

Theorem 6.9. *Minor-minimal non-GF(4)-representable matroids have rank and corank at most 4.*

We can now prove the main theorem.

Theorem 6.10. *There is some matrix C' representing M' if and only if, up to permuting rows and columns, A and A^T have no submatrix in the following set, where x, y, z are distinct non-zero elements of GF(4).*

$$\begin{aligned} & \begin{bmatrix} x & y & z \end{bmatrix}, \begin{bmatrix} x & y \\ 0 & x \end{bmatrix}, \begin{bmatrix} x & y \\ y & x \end{bmatrix}, \begin{bmatrix} x & x \\ y & z \end{bmatrix}, \begin{bmatrix} x & y \\ z & x \end{bmatrix}, \begin{bmatrix} x & x & 0 \\ x & 0 & x \end{bmatrix}, \begin{bmatrix} x & x & 0 \\ x & 0 & y \end{bmatrix}, \\ & \begin{bmatrix} x & x & 0 \\ y & 0 & y \end{bmatrix}, \begin{bmatrix} x & y & 0 \\ x & 0 & y \end{bmatrix}, \begin{bmatrix} x & 0 & 0 \\ 0 & y & z \end{bmatrix}, \begin{bmatrix} x & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & x \end{bmatrix}, \begin{bmatrix} x & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & y \end{bmatrix}, \begin{bmatrix} x & 0 & 0 \\ 0 & y & 0 \\ 0 & 0 & z \end{bmatrix}, \\ & \begin{bmatrix} x & y & x \\ y & y & 0 \\ x & 0 & 0 \end{bmatrix}, \begin{bmatrix} x & y & x \\ y & y & 0 \\ x & 0 & z \end{bmatrix}. \end{aligned}$$

Proof. It follows from Lemmas 6.5, 6.6, and 6.8 that A has no submatrix on the above list.

Conversely, suppose that the GF(4)-representable matroid M is chosen to be minimal subject to the property that the relaxation M' is not GF(4)-representable. Then M' has a minor N isomorphic to one of the excluded minors for the class of GF(4)-representable matroids. Assume that $N = M'/C \setminus D$ for some subsets C and D . If there is an element x in both D and the circuit-hyperplane X of M , then $M \setminus x = M' \setminus x$ by Lemma 2.2, so M also has an N -minor, contradicting the fact that M is GF(4)-representable. We deduce that $D \subseteq E(M) - X$, and dually, $C \subseteq X$. Now if $|D| \geq 2$, then there is some element x in both D and $E(M') - (X \cup f)$, so relaxing the circuit-hyperplane X of $M \setminus f$ gives $M' \setminus f$ that is not GF(4)-representable, which contradicts the minimality of M . Therefore $|D| \leq 1$, and by a dual argument, there is no element x in both C and $X - e$, so $|C| \leq 1$. Since we know, by Theorem 6.9, that $|E(N)| \leq 8$, it now follows that $|E(M')| \leq 10$. The computations in the Appendix below show that M' must have a submatrix from the above list. \square

REFERENCES

- [1] C. CHUN, D. CHUN, B. CLARK, D. MAYHEW, G. WHITTLE, AND S. H. M. VAN ZWAM, *Computer-verification of the structure of some classes of fragile matroids*, *arXiv:1312.5175*.

- [2] C. CHUN, D. CHUN, D. MAYHEW, AND S. H. M. VAN ZWAM, *Fan-extensions in fragile matroids*, Electron. J. Combin., 22 (2015).
- [3] B. CLARK, D. MAYHEW, G. WHITTLE, AND S. VAN ZWAM, *The structure of $\{U_{2,5}, U_{3,5}\}$ -fragile matroids*, SIAM Journal on Discrete Mathematics, 30 (2016), pp. 1480–1508.
- [4] J. GEELEN, A. GERARDS, AND A. KAPOOR, *The excluded minors for $GF(4)$ -representable matroids*, Journal of Combinatorial Theory, Series B, 79 (2000), pp. 247 – 299.
- [5] J. GEELEN, J. OXLEY, D. VERTIGAN, AND G. WHITTLE, *Weak maps and stabilizers of classes of matroids*, Adv. Appl. Math., 21 (1998), pp. 305–341.
- [6] F. JAEGER, D. L. VERTIGAN, AND D. J. WELSH, *On the computational complexity of the Jones and Tutte polynomials*, Math. Proc. Cambridge Philos. Soc, 108 (1990), pp. 35–53.
- [7] J. KAHN, *A problem of P. Seymour on nonbinary matroids*, Combinatorica, 5 (1985), pp. 319–323.
- [8] D. LUCAS, *Weak maps of combinatorial geometries*, Trans. Amer. Math. Soc., 206 (1975), pp. 247–279.
- [9] D. MAYHEW, S. H. M. VAN ZWAM, AND G. WHITTLE, *Stability, fragility, and Rota’s Conjecture*, J. Combin. Theory Ser. B, 102 (2012), pp. 760–783.
- [10] J. OXLEY, *Matroid theory, Second Edition*, Oxford University Press, New York, 2011.
- [11] J. OXLEY, C. SEMPLE, AND D. VERTIGAN, *Generalized Δ -Y exchange and k -regular matroids*, J. Combin. Theory Ser. B, 79 (2000), pp. 1–65.
- [12] J. OXLEY AND G. WHITTLE, *On weak maps of ternary matroids*, European J. Combin., 19 (1998), pp. 377–389.
- [13] R. PENDAVINGH AND S. H. M. VAN ZWAM, *Confinement of matroid representations to subsets of partial fields*, J. Combin. Theory Ser. B, 100 (2010), pp. 510–545.
- [14] C. SEMPLE AND G. WHITTLE, *On representable matroids having neither $U_{2,5}$ - nor $U_{3,5}$ -minors*, in Matroid Theory: AMS-IMS-SIAM Joint Summer Research Conference on Matroid Theory, July 2-6, 1995, University of Washington, Seattle, vol. 197, American Mathematical Soc., 1996, p. 377.
- [15] K. TRUEMPER, *Alpha-balanced graphs and matrices and $GF(3)$ -representability of matroids*, Journal of Combinatorial Theory, Series B, 32 (1982), pp. 112–139.

GF4-CHrelaxations

Small quaternary matroids with a circuit-hyperplane such that the relaxation is again quaternary

Preliminaries

```
#auto
from sage.matroids.advanced import *
GF4 = GF(4, 'a')
a = GF4.gen()
b = 1+a
```

Part I. Excluded submatrices.

Consider a quaternary matroid M with a circuit-hyperplane H such that the relaxation of H yields another quaternary matroid, M' . Let B be a basis of M intersecting H in $r - 1$ elements. Let e be the remaining element of the hyperplane. After applying row and column scaling, we may assume that the reduced representation matrices look as follows:

$$F = \begin{bmatrix} & 1 \\ A & \vdots \\ & 1 \\ 1 \cdots 1 & 0 \end{bmatrix} \text{ and } F' = \begin{bmatrix} & 1 \\ A' & \vdots \\ & 1 \\ 1 \cdots 1 & \alpha \end{bmatrix}$$

Our goal is to prove the following:

Lemma I. *Let M and A be constructed as above. Then M' is quaternary if and only if neither A nor A^T has a submatrix in the following set, where x, y, z are distinct, arbitrary non-zero elements:*

$$[1, \alpha, \bar{\alpha}], \begin{bmatrix} x & y \\ 0 & x \end{bmatrix}, \begin{bmatrix} x & y \\ y & x \end{bmatrix}, \begin{bmatrix} x & y \\ z & x \end{bmatrix}, \begin{bmatrix} x & x \\ y & z \end{bmatrix}, \begin{bmatrix} x & x & 0 \\ x & 0 & x \end{bmatrix}, \begin{bmatrix} x & x & 0 \\ x & 0 & y \end{bmatrix}, \begin{bmatrix} x & x & 0 \\ y & 0 & y \end{bmatrix}, \begin{bmatrix} x & y & 0 \\ x & 0 & y \end{bmatrix}, \\ \begin{bmatrix} x & y & 0 \\ x & 0 & z \end{bmatrix}, \begin{bmatrix} x & 0 & 0 \\ 0 & x & 0 \end{bmatrix}, \begin{bmatrix} x & 0 & 0 \\ 0 & 0 & y \end{bmatrix}, \begin{bmatrix} x & 0 & 0 \\ 0 & 0 & z \end{bmatrix}, \begin{bmatrix} x & y & x \\ y & y & 0 \end{bmatrix}, \begin{bmatrix} x & y & x \\ y & y & 0 \end{bmatrix}.$$

To complete the proof, we check all such M and A with $|M| \leq 10$. Carrying out this finite check needs a little bit of care; the naive approach of generating all $\text{GF}(4)$ -matrices gets too slow beyond size 3×3 . Instead, we will be generating these row by row, using submatrices to eliminate any new rows that need to be considered. Even so, these computations take over 8 hours to complete.

```
# Excluded minors for GF(4)-representable matroids:
ExMinGF4 = [matroids.Uniform(2,6), matroids.Uniform(4,6),
matroids.named_matroids.P6(), matroids.named_matroids.NonFano(),
matroids.named_matroids.NonFano().dual(), matroids.named_matroids.P8(),
matroids.named_matroids.P8pp()]

def is_relaxation_quaternary(A):
    """
    Tests if the circuit-hyperplane relaxation of the matroid associated
    with A (M[I F]) has an excluded minor isomorphic to one of the GF(4)-
    excluded minors
    """
    C = Matrix(GF4, A.nrows(), 1)
    for r in range(A.nrows()):
        C[r,0] = 1
    B = A.augment(C)
    C = Matrix(GF4, 1, B.ncols())
    for c in range(B.ncols()-1):
        C[0,c] = 1
    F = B.stack(C)
    M = Matroid(reduced_matrix=F)
    CH = M.groundset_list()[0:(F.nrows()-1)] + [M.groundset_list()[-1]]
    B = list(M.bases())
    B.append(frozenset(CH))
    M2 = Matroid(bases=B)
    return not any(M2.has_minor(N) for N in ExMinGF4)
```

We start by treating the $1 \times n$ matrices A separately. We generate all of them, and save them into a list A_rows . Then we test if their relaxation is quaternary by an excluded-minor check, and save the indices of the ones that survive in a list $legits[(1,n)]$. The others go into a list called $rejects[(1,n)]$. Next, we fill a list $legits$ with $r \times n$ matrices, encoded as a list of indices into A_rows . For each new row, we use the matrices in $legits$ one level below to filter which ones have no proper submatrices that were excluded. The new ones are then tested for whether their relaxation is quaternary, and moved into $legits$ or $rejects$ accordingly. To speed things up, we assume the rows of our matrices are sorted in ascending order with respect to the aforementioned indices.

```
A_rows = {0:[]}
legits = {(0,0):[tuple([])], (0,1):[tuple([])]}
rejects = {}
GF4elts = [GF4(0), GF4(1), a, b]
```



```

def make_valid_rows(n):
    A_rows[n] = []
    legits[(1,n)] = []
    rejects[(1,n)] = []

    for i in range(4^n):
        k = i
        v = [GF4elts[0]]*n # row vector to be filled
        for r in range(n):
            v[r] = GF4elts[mod(k,4)]
            k = k // 4
        A_rows[n].append(tuple(v))
        if is_relaxation_quaternary(Matrix(GF4, 1, n , v)):
            legits[(1,n)].append(tuple([len(A_rows[n])-1]))
        else:
            rejects[(1,n)].append(tuple([len(A_rows[n])-1]))

```

```

def rowlist_to_matrix(L, n):
    A = Matrix(GF4, len(L), n)
    for r in range(len(L)):
        for c in range(n):
            A[r,c] = A_rows[n][L[r]][c]
    return A

```

```

def matrix_to_rowlist(A):
    L = []
    n = A.ncols()
    for r in A.rows():
        v = tuple(r)
        L.append(A_rows[n].index(v))
    return tuple(sorted(L))

```

```

def contains_reject(A):
    r = A.nrows()
    c = A.ncols()
    for ci in range(1,c):
        for S in Subsets(range(c),ci):
            if r <= ci:
                Ap = A[:,list(S)]
                rj = r
                cj = ci
            else:
                Ap = A[:,list(S)].transpose()

```

```

        rj = ci
        cj = r
        if matrix_to_rowlist(Ap) in rejects[(rj,cj)]:
            return True
    return False

def add_row(r,n):
    """
    Assuming A_rows[n], legits[(r-1,n)], and rejects[(r-1,n)] are
    complete, generate legits[(r,n)] and rejects[(r,n)].
    """
    legits[(r,n)] = []
    rejects[(r,n)] = []
    for A1 in legits[(r-1,n)]:
        for v in legits[(1,n)]:
            if v[0] >= A1[-1]: # row vectors are ordered.
                # check if all submatrices, obtained by removing a row,
                occur as valid.
                contains_smaller_reject = False
                for i in range(r-1):
                    if (A1[0:i] + A1[i+1:r-1] + v) not in legits[(r-
1,n)]:
                        contains_smaller_reject = True
                        break
                if not contains_smaller_reject and not
contains_reject(rowlist_to_matrix(A1 + v, n)):
                    if is_relaxation_quaternary(rowlist_to_matrix(A1 +
v, n)):
                        legits[(r,n)].append(A1+v)
                    else:
                        rejects[(r,n)].append(A1+v)

```

```

make_valid_rows(1)
print legits
print rejects

```

```

{(0, 1): [()], (0, 0): [()], (1, 1): [(0,), (1,), (2,), (3,)]}
{(1, 1): []}

```

```

make_valid_rows(2)
add_row(2,2)
for L in rejects[(2,2)]:
    print rowlist_to_matrix(L,2)
    print ""

```

```

[1 0]
[a 1]

```

$$\begin{bmatrix} 1 & 0 \\ a + 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} a & 0 \\ 1 & a \end{bmatrix}$$

$$\begin{bmatrix} a & 0 \\ a + 1 & a \end{bmatrix}$$

$$\begin{bmatrix} a + 1 & 0 \\ 1 & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} a + 1 & 0 \\ a & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 1 \\ 1 & a \end{bmatrix}$$

$$\begin{bmatrix} 0 & 1 \\ 1 & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 \\ a + 1 & a \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 \\ a & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} a & 1 \\ a + 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} a & 1 \\ 0 & a \end{bmatrix}$$

$$\begin{bmatrix} a & 1 \\ 1 & a \end{bmatrix}$$

$$\begin{bmatrix} a & 1 \\ a + 1 & a \end{bmatrix}$$

$$\begin{bmatrix} a & 1 \\ 1 & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} a & 1 \\ a & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} a & 1 \\ a + 1 & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} a + 1 & 1 \\ 1 & a \end{bmatrix}$$

$$\begin{bmatrix} a + 1 & 1 \\ a & a \end{bmatrix}$$

$$\begin{bmatrix} a + 1 & 1 \\ a + 1 & a \end{bmatrix}$$

$$\begin{bmatrix} a + 1 & 1 \\ 0 & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} a + 1 & 1 \\ 1 & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} a + 1 & 1 \\ a & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & a \\ a & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & a \\ a + 1 & a \end{bmatrix}$$

$$\begin{bmatrix} 1 & a \\ 1 & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & a \\ a & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & a \\ a + 1 & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} a & a \\ 1 & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} a + 1 & a \\ 0 & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} a + 1 & a \\ 1 & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} a + 1 & a \\ a & a + 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & a + 1 \\ a & a + 1 \end{bmatrix}$$

```

make_valid_rows(3)
add_row(2,3)
print len(rejects[(2,3)])
add_row(3,3)
print len(rejects[(3,3)])

```

```
117
```

```
99
```

```

make_valid_rows(4)
add_row(2,4)
print len(rejects[(2,4)])
add_row(3,4)
print len(rejects[(3,4)])
add_row(4,4)
print len(rejects[(4,4)])

```

```
0
```

```
0
```

```
0
```

```

save(A_rows, "/Users/svanzwam/Dropbox/A_rows.sobj")
save(legits, "/Users/svanzwam/Dropbox/legits.sobj")
save(rejects, "/Users/svanzwam/Dropbox/rejects.sobj")

```

We condense the information gained as follows. Each forbidden submatrix gives rise to several entries in our `rejects` list through column permutations, scaling the entire matrix by a nonzero constant, and through applying a field automorphism followed by zero or more of the above. The encoding of the matrices as a list of indices into `A_rows` gives a natural ordering. We determine and print the lexicographically least of each equivalence class.

```

def Abar(A):
    """
    Applies the nontrivial GF(4)-automorphism to each entry of matrix A
    """
    B = copy(A)
    for i in range(B.nrows()):
        for j in range(B.ncols()):
            B[i,j] = B[i,j]^2
    return B

def canonical_rep_matrix(L, n):
    """
    INPUT:

    - L -- a matrix A, represented as a tuple of row indices
    - n -- the number of columns in A
    """

```

OUTPUT:

- B -- a matrix, represented as a tuple of row indices, equivalent to A under the operations described above, and with minimum vector of indices

"""

```
Am = rowlist_to_matrix(L,n)
Bm = Abar(Am)
starters = [Am, a*Am, b*Am, Bm, a*Bm, b*Bm]
candidates = []
for X in starters:
    for P in Permutations(range(n)):
        candidates.append(matrix_to_rowlist(X[:,list(P)]))
return min(candidates)
```

```
A = Matrix(GF4, 2,2, [0,a,1,a+1])
print A
L = matrix_to_rowlist(A)
print L
```

```
[ 0 a]
[ 1 a + 1]
(8, 13)
```

```
K = canonical_rep_matrix(L,2)
print K
print rowlist_to_matrix(K,2)
```

```
(1, 11)
[ 1 0]
[a + 1 a]
```

```
rejects_minimal = {}
rejects_dropped = {}
for X in [(2,2),(1,3),(2,3),(3,3)]:
    rejects_minimal[X] = []
    rejects_dropped[X] = []
    for L in rejects[X]:
        K = canonical_rep_matrix(L,X[1])
        if L == K:
            rejects_minimal[X].append(L)
        else:
            rejects_dropped[X].append(K)
```

```
for X in [(2,2), (1,3), (2,3), (3,3)]:
    print "Statistics for size ", X
```

```

    print len(rejects[X]), " should be equal to ",
len(rejects_minimal[X]), "+", len(rejects_dropped[X]), "=",
len(rejects_minimal[X])+len(rejects_dropped[X])
    for L in rejects_dropped[X]:
        if not L in rejects_minimal[X]:
            # This should never happen
            print "The following canonical representative is absent from
our list: ", L
        for L in rejects_minimal[X]:
            print rowlist_to_matrix(L,X[1])
            print " "

```

```

Statistics for size (2, 2)
33 should be equal to 5 + 28 = 33
[1 0]
[a 1]

```

```

[ 1 1]
[a + 1 a]

```

```

[ a 1]
[a + 1 1]

```

```

[a 1]
[1 a]

```

```

[ a 1]
[a + 1 a]

```

```

Statistics for size (1, 3)
6 should be equal to 1 + 5 = 6
[a + 1 a 1]

```

```

Statistics for size (2, 3)
117 should be equal to 6 + 111 = 117
[ 1 0 0]
[ 0 a + 1 a]

```

```

[1 1 0]
[1 0 1]

```

```

[1 1 0]
[1 0 a]

```

```

[1 1 0]
[a 0 a]

```

```

[a 1 0]

```

```
[a 0 1]
```

```
[      a      1      0]
[      a      0 a + 1]
```

```
Statistics for size (3, 3)
99 should be equal to 5 + 94 = 99
```

```
[1 0 0]
[0 1 0]
[0 0 1]
```

```
[1 0 0]
[0 1 0]
[0 0 a]
```

```
[      1      0      0]
[      0      a      0]
[      0      0 a + 1]
```

```
[1 0 0]
[a a 0]
[1 a 1]
```

```
[      1      1      0]
[a + 1      0      a]
[a + 1      1 a + 1]
```

This concludes the proof of the lemma.

Part II. Structure.

We study the quaternary, $\{U_{2,5}, U_{3,5}\}$ -fragile matroids with no minor isomorphic to a matroid in $\{P_8^-, F_7^-, (F_7^-)^*\}$.

```
A = Matrix(GF4, [[1,1,1],[1,a,b]])
U25 = Matroid(reduced_matrix=A)
U25
```

Quaternary matroid of rank 2 on 5 elements

```
def has_U25_or_U35(M):
    d = 1
    while d > 0:
        N = M.simplify().cosimplify()
        d = len(M) - len(N)
        M = N
```



```

    if M.full_corank() == 2:
        return len(M) >= 5
    return M.has_line_minor(5)

def is_fragile(M):
    for e in M.groundset():
        if has_U25_or_U35(M \ e) and has_U25_or_U35(M / e):
            return False
    return True

```

```

# P8m is a circuit-hyperplane relaxation of P8, obtained by relaxing the
top or bottom plane of the twisted cube.
P8m1 = matroids.named_matroids.P8()
P8m = Matroid(groundset='abcdefgh', field=GF4, reduced_matrix=
[[b,1,1,0],[1,1,0,1],[1,0,b,1],[0,1,a,a]])
setprint([B for B in P8m.bases() if not B in P8m1.bases()])
setprint([B for B in P8m1.bases() if not B in P8m.bases()])

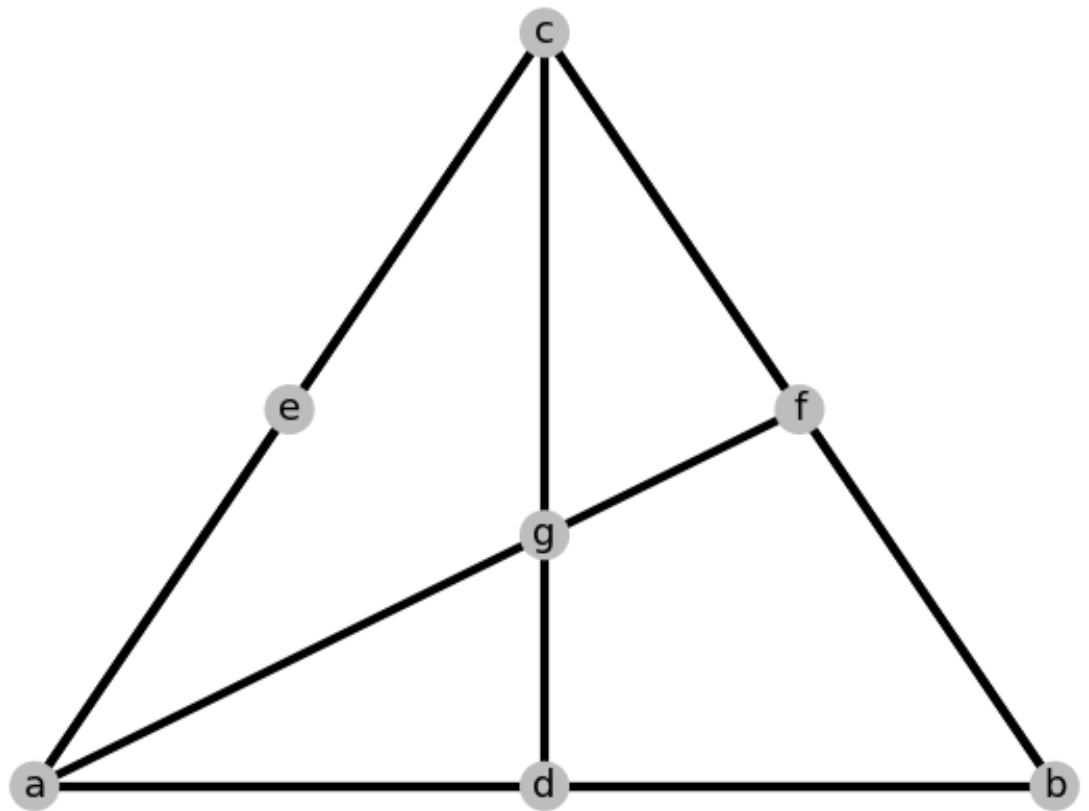
[{'b', 'c', 'f', 'g'}]
[]

```

```

# F7mm is obtained from F7 by relaxing two lines.
F7mm = Matroid(groundset='abcdefg', field=GF4, reduced_matrix=
[[1,1,0,1],[1,0,1,1],[0,a,1,1]])
F7mm.show()
F7mmd = F7mm.dual()

```



Generation of the 3-connected matroids in the class with at most 9 elements

We will generate the $\{U_{2,5}, U_{3,5}\}$ -fragile, $\text{GF}(4)$ -representable, 3-connected matroids with at most 9 elements. By the Splitter Theorem, then, we can generate all these matroids by repeatedly extending or coextending by a single element. The following method does this.

We assume that the elements are labeled $\{0, 1, \dots, n-1\}$, and will use n for the label of the new element.

```
def plus1(X):
    res = []
    for M in X:
        res.extend(M.linear_extensions(element=len(M), simple=True))
```

```

        res.extend(M.linear_coextensions(element=len(M),
                                           cosimple=True))
    res = [M for M in res if not any(M.has_minor(N) for N in [P8m, F7mm,
F7mmd]) and is_fragile(M)]
    res = get_nonisomorphic_matroids(res)
    return res

```

```

%time
cat = {}
cat[5] = [U25, U25.dual()]
print "Number of 5-element matroids: ", len(cat[5])
cat[6] = plus1(cat[5])
print "Number of 6-element matroids: ", len(cat[6])
cat[7] = plus1(cat[6])
print "Number of 7-element matroids: ", len(cat[7])
cat[8] = plus1(cat[7])
print "Number of 8-element matroids: ", len(cat[8])
cat[9] = plus1(cat[8])
print "Number of 9-element matroids: ", len(cat[9])

```

```

Number of 5-element matroids: 2
Number of 6-element matroids: 1
Number of 7-element matroids: 4
Number of 8-element matroids: 8
Number of 9-element matroids: 20
CPU time: 5.86 s, Wall time: 5.99 s

```

Note that different runs of this generation code are not guaranteed to produce the matroids in the same order (though the sets will always be the same). In order to make the code below reproducible, we sort the lists. Through experimentation, we learned that the tuple (r, b, c, d) , where r is the rank, b is the number of bases, c is the number of circuits, and d is the number of cocircuits, uniquely identifies each matroid in our class up to nine elements.

```

def f(M):
    return (M.rank(), M.bases_count(), len(M.circuits()),
len(M.cocircuits()))

for i in cat:
    cat[i] = sorted(cat[i], key=f)

```

We go through the matroids generated, printing the circuit closures for easy cross referencing to the figures from [1].

```

cat[5][0].representation(reduced=True, B=[0,1], labels=False)

[ 1 1 1]

```

```
[ 1 a a + 1]
```

```
setprint(cat[6][0].circuit_closures())
cat[6][0].is_isomorphic(matroids.named_matroids.Q6())
{2: {{0, 1, 4}, {2, 3, 4}}, 3: {{0, 1, 2, 3, 4, 5}}}
True
```

```
# M7,1
setprint(cat[7][1].circuit_closures())
print cat[7][3].dual().is_isomorphic(cat[7][1])
{2: {{0, 1, 4}, {0, 3, 6}, {2, 3, 4}, {2, 5, 6}}, 3: {{0, 1, 2, 3, 4, 5, 6}}}
True
```

```
# M7,0
setprint(cat[7][0].circuit_closures())
print cat[7][2].dual().is_isomorphic(cat[7][0])
{2: {{0, 1, 4, 6}, {3, 5, 6}, {2, 3, 4}}, 3: {{0, 1, 2, 3, 4, 5, 6}}}
True
```

```
# Y8
setprint(cat[8][0].circuit_closures())
print cat[8][7].dual().is_isomorphic(cat[8][0])
{2: {{1, 2, 7}, {0, 1, 4, 6}, {2, 3, 4}, {3, 5, 6, 7}}, 3: {{0, 1, 2, 3, 4, 5, 6, 7}}}
True
```

```
setprint(cat[8][1].circuit_closures())
print cat[8][1].dual().is_isomorphic(cat[8][1])
{2: {{0, 1, 4}, {3, 5, 6}, {2, 3, 4}}, 3: {{2, 3, 4, 5, 6}, {0, 1, 2, 3, 4}, {0, 1, 4, 6, 7}}, 4: {{0, 1, 2, 3, 4, 5, 6, 7}}}
True
```

```
# X8
setprint(cat[8][2].circuit_closures())
print cat[8][2].dual().is_isomorphic(cat[8][2])
{2: {{0, 1, 4, 6}}, 3: {{0, 2, 3, 5}, {2, 3, 4, 7}, {3, 5, 6, 7}}, 4: {{0, 1, 2, 3, 4, 5, 6, 7}}}
True
```

```
setprint(cat[8][3].circuit_closures())
print cat[8][3].dual().is_isomorphic(cat[8][4])
setprint(cat[8][4].circuit_closures())
{2: {{0, 4, 6}, {3, 5, 6}}, 3: {{1, 2, 3, 5, 6}, {0, 3, 4, 5, 6}, {0, 1, 4, 6, 7}, {2, 3, 4, 7}}, 4: {{0, 1, 2, 3, 4, 5, 6, 7}}}
True
{2: {{0, 1, 4}, {2, 3, 4}, {2, 5, 6}}, 3: {{2, 3, 4, 5, 6}, {0, 1, 2, 3, 4}, {0, 3, 6, 7}}, 4: {{0, 1, 2, 3, 4, 5, 6, 7}}}
```

```

setprint(cat[8][5].circuit_closures())
print cat[8][5].dual().is_isomorphic(cat[8][5])
{2: {{0, 1, 4}, {3, 5, 6}}, 3: {{1, 2, 3, 5, 6}, {0, 1, 4, 6, 7},
{2, 3, 4, 7}}, 4: {{0, 1, 2, 3, 4, 5, 6, 7}}}
True

```

```

setprint(cat[8][6].circuit_closures())
print cat[8][6].dual().is_isomorphic(cat[8][6])
{2: {{0, 1, 4}, {2, 3, 4}}, 3: {{1, 2, 5, 6}, {0, 1, 2, 3, 4}, {0,
1, 4, 6, 7}, {3, 5, 6, 7}}, 4: {{0, 1, 2, 3, 4, 5, 6, 7}}}
True

```

```

X8 = cat[8][2]
Y8 = cat[8][0]
Y8d = cat[8][7]

```

```

print cat[9][0].is_isomorphic(cat[9][10].dual())
print cat[9][1].is_isomorphic(cat[9][11].dual())
print cat[9][2].is_isomorphic(cat[9][12].dual())
print cat[9][3].is_isomorphic(cat[9][13].dual())
print cat[9][4].is_isomorphic(cat[9][14].dual())
print cat[9][5].is_isomorphic(cat[9][15].dual())
print cat[9][6].is_isomorphic(cat[9][16].dual())
print cat[9][7].is_isomorphic(cat[9][17].dual())
print cat[9][8].is_isomorphic(cat[9][18].dual())
print cat[9][9].is_isomorphic(cat[9][19].dual())

```

```

True
True
True
True
True
True
True
True
True
True

```

```

for i in range(10):
    print "M_9,", i
    setprint(cat[9][i].circuit_closures())

```

```

M_9, 0
{2: {{0, 1, 4, 8}, {3, 5, 6}, {2, 3, 4}, {6, 7, 8}}, 3: {{2, 3, 4,
5, 6}, {3, 5, 6, 7, 8}, {0, 1, 2, 3, 4, 8}, {0, 1, 4, 6, 7, 8}}, 4:
{{0, 1, 2, 3, 4, 5, 6, 7, 8}}}
M_9, 1
{2: {{2, 3, 8}, {0, 4, 6}, {1, 6, 8}, {3, 5, 6}, {4, 7, 8}}, 3: {{2,
3, 4, 7, 8}, {1, 2, 3, 5, 6, 8}, {0, 3, 4, 5, 6}, {0, 1, 4, 6, 7,

```

```

8}}, 4: {{0, 1, 2, 3, 4, 5, 6, 7, 8}}}
M_9, 2
{2: {{0, 1, 4}, {1, 6, 8}, {3, 5, 6}, {2, 3, 8}, {4, 7, 8}}, 3: {{2,
3, 4, 7, 8}, {1, 2, 3, 5, 6, 8}, {0, 1, 4, 6, 7, 8}}, 4: {{0, 1, 2,
3, 4, 5, 6, 7, 8}}}
M_9, 3
{2: {{1, 2, 7}, {0, 1, 4, 6}, {3, 5, 6}}, 3: {{0, 1, 2, 4, 6, 7},
{3, 5, 6, 7, 8}, {0, 1, 3, 4, 5, 6}, {2, 3, 4, 8}}, 4: {{0, 1, 2, 3,
4, 5, 6, 7, 8}}}
M_9, 4
{2: {{0, 1, 4, 6}, {2, 3, 4}, {3, 5, 7}}, 3: {{1, 2, 7, 8}, {2, 3,
4, 5, 7}, {0, 1, 2, 3, 4, 6}, {3, 5, 6, 7, 8}}, 4: {{0, 1, 2, 3, 4,
5, 6, 7, 8}}}
M_9, 5
{2: {{0, 1, 4}, {1, 7, 8}, {3, 5, 6}, {2, 3, 4}, {0, 6, 8}}, 3: {{2,
3, 4, 5, 6}, {0, 1, 2, 3, 4}, {0, 1, 4, 6, 7, 8}, {0, 3, 5, 6, 8}},
4: {{0, 1, 2, 3, 4, 5, 6, 7, 8}}}
M_9, 6
{2: {{1, 2, 7}, {2, 3, 4}, {3, 5, 7}, {0, 1, 6}}, 3: {{0, 1, 2, 6,
7}, {1, 2, 3, 4, 5, 7}, {0, 1, 4, 6, 8}, {3, 5, 6, 7, 8}}, 4: {{0,
1, 2, 3, 4, 5, 6, 7, 8}}}
M_9, 7
{2: {{0, 1, 4}, {3, 5, 8}, {2, 3, 4}, {6, 7, 8}}, 3: {{1, 2, 5, 6},
{3, 5, 6, 7, 8}, {0, 1, 2, 3, 4}, {0, 1, 4, 6, 7, 8}, {2, 3, 4, 5,
8}}, 4: {{0, 1, 2, 3, 4, 5, 6, 7, 8}}}
M_9, 8
{2: {{0, 1, 4}, {3, 7, 8}, {2, 3, 4}, {0, 6, 8}, {2, 5, 6}}, 3: {{2,
3, 4, 5, 6}, {0, 1, 2, 3, 4}, {0, 2, 5, 6, 8}, {2, 3, 4, 7, 8}, {0,
1, 4, 6, 8}, {0, 3, 6, 7, 8}}, 4: {{0, 1, 2, 3, 4, 5, 6, 7, 8}}}
M_9, 9
{2: {{0, 1, 4}, {5, 6, 8}, {3, 7, 8}, {2, 3, 4}}, 3: {{2, 3, 4, 7,
8}, {3, 5, 6, 7, 8}, {0, 1, 2, 3, 4}, {1, 2, 5, 6, 8}, {0, 1, 4, 6,
7}}, 4: {{0, 1, 2, 3, 4, 5, 6, 7, 8}}}

```

Our computation, at first, applies to the matroids with no minor in $\{X_8, Y_8, Y_8^*\}$, as well as no minor in $\{F_7^=, (F_7^=)^*, P_8^-\}$. That rules out the following:

```

[i for i in range(10) if cat[9][i].has_minor(X8) or
                        cat[9][i].has_minor(Y8) or
                        cat[9][i].has_minor(Y8d)]

```

```

[3, 4, 6]

```

```

excluded_minors = [P8m, F7mm, F7mmd, X8, Y8, Y8d]

```

From now, let \mathcal{M} be the class of $\{U_{2,5}, U_{3,5}\}$ -fragile quaternary matroids with no minor in $\{X_8, Y_8, Y_8^*, F_7^=, (F_7^=)^*, P_8^-\}$.

Lemma 1. *The matroid $M_{9,9}$ is a splitter for \mathcal{M} .*

```
print [M for M in cat[9][9].linear_extensions(simple=True)
      if is_fragile(M) and not any(M.has_minor(N) for N in
excluded_minors)]
print [M for M in cat[9][9].linear_coextensions(cosimple=True)
      if is_fragile(M) and not any(M.has_minor(N) for N in
excluded_minors)]

[]
[]
```

```
def print_fans(M):
    print "Triangles: "
    setprint([C for C in M.circuits() if len(C) == 3])
    print "Triads: "
    setprint([C for C in M.cocircuits() if len(C) == 3])
```

Consider the following set of three disjoint triads of $M_{9,18}$: $\mathcal{F} = \{(2, 3, 5), (7, 4, 0), (1, 6, 8)\}$, where the ordering of the elements was chosen so the deletable element is in the center. We will show:

Lemma 2. *Every 3-connected matroid in \mathcal{M} with a minor isomorphic to $M_{9,18}$ is a fan-extension of $M_{9,18}$ with respect to \mathcal{F} .*

```
M18 = cat[9][18]
print_fans(M18)
print "Deletable elements: "
setprint([e for e in M18.groundset() if (M18 \ e).has_minor(U25)])

Triangles:
[]
Triads:
[{0, 5, 8}, {1, 6, 8}, {2, 3, 5}, {1, 2, 7}, {0, 4, 7}]
Deletable elements:
[3, 4, 6]
```

```
exList = [M for M in M18.linear_extensions(9, simple=True)
          if is_fragile(M) and not any(M.has_minor(N) for N in
excluded_minors)]
exList.extend([M for M in M18.linear_coextensions(9, cosimple=True)
              if is_fragile(M) and not any(M.has_minor(N) for N in
excluded_minors)])
print len(exList)
```

2

```
for M in exList:
    print M.rank()
    print_fans(M)
```

```
print " "
```

```
5
```

```
Triangles:
```

```
[{4, 7, 9}, {1, 6, 9}, {2, 3, 9}]
```

```
Triads:
```

```
[{0, 4, 7}, {2, 3, 5}, {1, 6, 8}, {0, 5, 8}]
```

```
5
```

```
Triangles:
```

```
[{6, 8, 9}, {3, 5, 9}, {0, 4, 9}]
```

```
Triads:
```

```
[{1, 2, 7}, {0, 4, 7}, {2, 3, 5}, {1, 6, 8}]
```

```
doubleList = []
```

```
for N in exList:
```

```
    doubleList.extend([M for M in N.linear_extensions(10,
                                                         simple=True)
                       if is_fragile(M) and not any(M.has_minor(N) for N
```

```
in excluded_minors)])
```

```
    doubleList.extend([M for M in N.linear_coextensions(10,
                                                         cosimple=True)
                       if is_fragile(M) and not any(M.has_minor(N) for N
```

```
in excluded_minors)])
```

```
print len(doubleList)
```

```
8
```

```
for M in doubleList:
```

```
    print M.rank()
```

```
    print_fans(M)
```

```
    print " "
```

```
5
```

```
Triangles:
```

```
[{4, 7, 9}, {1, 6, 9}, {2, 3, 9}, {6, 8, 10}, {3, 5, 10}, {0, 4, 10}]
```

```
Triads:
```

```
[{1, 6, 8}, {2, 3, 5}, {0, 4, 7}]
```

```
6
```

```
Triangles:
```

```
[{4, 7, 9}]
```

```
Triads:
```

```
[{7, 9, 10}, {1, 2, 10}, {0, 5, 8}, {1, 6, 8}, {2, 3, 5}, {0, 4, 7}]
```

```
6
```

```
Triangles:
```

```
[{2, 3, 9}]
```

```
Triads:
```



```

[{{2, 9, 10}, {1, 7, 10}, {0, 5, 8}, {1, 6, 8}, {2, 3, 5}, {0, 4, 7}}]

6
Triangles:
[{{1, 6, 9}}]
Triads:
[{{1, 9, 10}, {2, 7, 10}, {0, 5, 8}, {1, 6, 8}, {2, 3, 5}, {0, 4, 7}}]

5
Triangles:
[{{6, 8, 9}, {3, 5, 9}, {0, 4, 9}, {4, 7, 10}, {1, 6, 10}, {2, 3, 10}}]
Triads:
[{{1, 6, 8}, {2, 3, 5}, {0, 4, 7}}]

6
Triangles:
[{{0, 4, 9}}]
Triads:
[{{0, 9, 10}, {5, 8, 10}, {1, 6, 8}, {2, 3, 5}, {0, 4, 7}, {1, 2, 7}}]

6
Triangles:
[{{6, 8, 9}}]
Triads:
[{{8, 9, 10}, {0, 5, 10}, {1, 6, 8}, {2, 3, 5}, {0, 4, 7}, {1, 2, 7}}]

6
Triangles:
[{{3, 5, 9}}]
Triads:
[{{5, 9, 10}, {0, 8, 10}, {1, 6, 8}, {2, 3, 5}, {0, 4, 7}, {1, 2, 7}}]

```

All are fan-extensions, which concludes the proof of the lemma.

Lemma 3. *Every 3-connected matroid in \mathcal{M} with an $M_{9,7}$ -minor but no minor in $\{X_8, Y_8, Y_8^*\}$ is a fan-extension of $M_{9,7}$ with respect to $\mathcal{F} = \{(4, 2, 3, 5, 8)\}$.*

```

M7 = cat[9][7]
print_fans(M7)

```

```

Triangles:
[{{0, 1, 4}, {2, 3, 4}, {6, 7, 8}, {3, 5, 8}}]
Triads:
[{{2, 3, 5}}]

```

```

exList = [M for M in M7.linear_extensions(9, simple=True)
           if is_fragile(M) and not any(M.has_minor(N) for N in

```

```

excluded_minors)]
exList.extend([M for M in M7.linear_coextensions(9, cosimple=True)
               if is_fragile(M) and not any(M.has_minor(N) for N in
excluded_minors)])
print len(exList)

```

2

```

for M in exList:
    print M.rank()
    print_fans(M)
    print " "

```

5

```

Triangles:
[{2, 3, 4}, {6, 7, 8}, {3, 5, 8}]
Triads:
[{0, 1, 9}, {2, 4, 9}, {2, 3, 5}]

```

5

```

Triangles:
[{2, 3, 4}, {0, 1, 4}, {3, 5, 8}]
Triads:
[{5, 8, 9}, {6, 7, 9}, {2, 3, 5}]

```

```

doubleList = []
for N in exList:
    doubleList.extend([M for M in N.linear_extensions(10,
                                                       simple=True)
                      if is_fragile(M) and not any(M.has_minor(N) for N
in excluded_minors)])
    doubleList.extend([M for M in N.linear_coextensions(10,
                                                         cosimple=True)
                      if is_fragile(M) and not any(M.has_minor(N) for N
in excluded_minors)])
print len(doubleList)

```

4

```

for M in doubleList:
    print M.rank()
    print_fans(M)
    print " "

```

5

```

Triangles:
[{2, 3, 4}, {6, 7, 8}, {3, 5, 8}, {4, 9, 10}, {0, 1, 10}]
Triads:
[{2, 3, 5}, {2, 4, 9}]

```

6

```

Triangles:
[{2, 3, 4}, {3, 5, 8}]
Triads:
[{6, 7, 10}, {5, 8, 10}, {2, 3, 5}, {2, 4, 9}, {0, 1, 9}]

5
Triangles:
[{2, 3, 4}, {0, 1, 4}, {3, 5, 8}, {8, 9, 10}, {6, 7, 10}]
Triads:
[{2, 3, 5}, {5, 8, 9}]

6
Triangles:
[{2, 3, 4}, {3, 5, 8}]
Triads:
[{2, 4, 10}, {0, 1, 10}, {2, 3, 5}, {6, 7, 9}, {5, 8, 9}]

```

All are fan-extensions, which concludes the proof of the lemma.

Lemma 4. *Every 3-connected matroid in \mathcal{M} with an $M_{9,15}$ -minor but no $M_{9,8}$ - or $M_{9,18}$ -minor is a fan-extension of $M_{9,15}$ with respect to $\mathcal{F} = \{(7, 4, 0, 6, 8), (2, 3, 5)\}$.*

```

M15 = cat[9][15]
print_fans(M15)

```

```

Triangles:
[{0, 4, 6}]
Triads:
[{1, 5, 8}, {0, 6, 8}, {2, 3, 5}, {1, 2, 7}, {0, 4, 7}]

```

```

M18d = M18.dual()
exList = [M for M in M15.linear_extensions(9, simple=True)
           if is_fragile(M) and not M.has_minor(M18)
           and not M.has_minor(M18d)
           and not any(M.has_minor(N) for N in
excluded_minors)]
exList.extend([M for M in M15.linear_coextensions(9, cosimple=True)
               if is_fragile(M) and not M.has_minor(M18)
               and not M.has_minor(M18d)
               and not any(M.has_minor(N) for N in
excluded_minors)])
print len(exList)

```

2

```

for M in exList:
    print M.rank()
    print_fans(M)
    print " "

```

```

5
Triangles:
[{0, 4, 6}, {6, 8, 9}, {3, 5, 9}]
Triads:
[{1, 2, 7}, {0, 4, 7}, {2, 3, 5}, {0, 6, 8}]

```

```

5
Triangles:
[{0, 4, 6}, {4, 7, 9}, {2, 3, 9}]
Triads:
[{0, 4, 7}, {2, 3, 5}, {0, 6, 8}, {1, 5, 8}]

```

```

doubleList = []
for N in exList:
    doubleList.extend([M for M in N.linear_extensions(10,
                                                         simple=True)
                       if is_fragile(M) and not M.has_minor(M18)
                           and not M.has_minor(M18d)
                           and not any(M.has_minor(N) for N
in excluded_minors)])
    doubleList.extend([M for M in N.linear_coextensions(10,
                                                         cosimple=True)
                       if is_fragile(M) and not M.has_minor(M18)
                           and not M.has_minor(M18d)
                           and not any(M.has_minor(N) for N
in excluded_minors)])
print len(doubleList)

```

6

```

for M in doubleList:
    print M.rank()
    print_fans(M)
    print " "

```

```

5
Triangles:
[{0, 4, 6}, {6, 8, 9}, {3, 5, 9}, {1, 9, 10}, {4, 7, 10}, {2, 3,
10}]
Triads:
[{0, 6, 8}, {2, 3, 5}, {0, 4, 7}]

```

```

6
Triangles:
[{0, 4, 6}, {6, 8, 9}]
Triads:
[{8, 9, 10}, {1, 5, 10}, {0, 6, 8}, {2, 3, 5}, {0, 4, 7}, {1, 2, 7}]

```

6

```

Triangles:
[{0, 4, 6}, {3, 5, 9}]
Triads:
[{5, 9, 10}, {1, 8, 10}, {0, 6, 8}, {2, 3, 5}, {0, 4, 7}, {1, 2, 7}]

5
Triangles:
[{0, 4, 6}, {4, 7, 9}, {2, 3, 9}, {1, 9, 10}, {6, 8, 10}, {3, 5, 10}]
Triads:
[{0, 6, 8}, {2, 3, 5}, {0, 4, 7}]

6
Triangles:
[{0, 4, 6}, {4, 7, 9}]
Triads:
[{7, 9, 10}, {1, 2, 10}, {1, 5, 8}, {0, 6, 8}, {2, 3, 5}, {0, 4, 7}]

6
Triangles:
[{0, 4, 6}, {2, 3, 9}]
Triads:
[{2, 9, 10}, {1, 7, 10}, {1, 5, 8}, {0, 6, 8}, {2, 3, 5}, {0, 4, 7}]

```

These are all fan-extensions, which concludes the proof of the lemma.

Lemma 5. *Every 3-connected matroid in M with an $M_{9,2}$ -minor is a fan-extension of $M_{9,2}$ with respect to $\mathcal{F} = \{(1, 0, 4, 7), (6, 5, 3, 2)\}$.*

Note that element 8 can be added to either fan.

```

M2 = cat[9][2]
print_fans(M2)

```

```

Triangles:
[{0, 1, 4}, {3, 5, 6}, {4, 7, 8}, {1, 6, 8}, {2, 3, 8}]
Triads:
[{2, 3, 5}, {0, 4, 7}]

```

```

exList = [M for M in M2.linear_extensions(9, simple=True)
           if is_fragile(M) and not any(M.has_minor(N) for N in
excluded_minors)]
exList.extend([M for M in M2.linear_coextensions(9, cosimple=True)
               if is_fragile(M) and not any(M.has_minor(N) for N in
excluded_minors)])
print len(exList)

```

3

```

for M in exList:
    print M.rank()
    print_fans(M)
    print " "

5
Triangles:
[{0, 1, 4}, {3, 5, 6}, {4, 7, 8}, {2, 3, 8}]
Triads:
[{5, 6, 9}, {0, 1, 9}, {0, 4, 7}, {2, 3, 5}]

5
Triangles:
[{3, 5, 6}, {0, 1, 4}, {4, 7, 8}]
Triads:
[{7, 8, 9}, {0, 4, 7}, {2, 3, 5}]

5
Triangles:
[{0, 1, 4}, {3, 5, 6}, {2, 3, 8}]
Triads:
[{2, 8, 9}, {0, 4, 7}, {2, 3, 5}]

```

```

doubleList = []
for N in exList:
    doubleList.extend([M for M in N.linear_extensions(10,
                                                         simple=True)
                       if is_fragile(M) and not any(M.has_minor(N) for N
in excluded_minors)])
    doubleList.extend([M for M in N.linear_coextensions(10,
                                                         cosimple=True)
                       if is_fragile(M) and not any(M.has_minor(N) for N
in excluded_minors)])
print len(doubleList)

```

8

```

for M in doubleList:
    print M.rank()
    print_fans(M)
    print " "

5
Triangles:
[{0, 1, 4}, {3, 5, 6}, {4, 7, 8}, {2, 3, 8}, {1, 9, 10}, {6, 8, 10}]
Triads:
[{2, 3, 5}, {0, 4, 7}, {0, 1, 9}]

5

```

Triangles:

[{0, 1, 4}, {3, 5, 6}, {4, 7, 8}, {2, 3, 8}, {6, 9, 10}, {1, 8, 10}]

Triads:

[{2, 3, 5}, {0, 4, 7}, {5, 6, 9}]

6

Triangles:

[{3, 5, 6}, {0, 1, 4}, {2, 3, 8}]

Triads:

[{7, 9, 10}, {2, 8, 10}, {2, 3, 5}, {0, 4, 7}, {0, 1, 9}, {5, 6, 9}]

6

Triangles:

[{3, 5, 6}, {0, 1, 4}, {4, 7, 8}]

Triads:

[{2, 9, 10}, {7, 8, 10}, {2, 3, 5}, {0, 4, 7}, {0, 1, 9}, {5, 6, 9}]

5

Triangles:

[{3, 5, 6}, {0, 1, 4}, {4, 7, 8}, {8, 9, 10}, {2, 3, 10}, {1, 6, 10}]

Triads:

[{0, 4, 7}, {2, 3, 5}, {7, 8, 9}]

6

Triangles:

[{3, 5, 6}, {0, 1, 4}, {4, 7, 8}]

Triads:

[{2, 9, 10}, {0, 1, 10}, {5, 6, 10}, {2, 3, 5}, {0, 4, 7}, {7, 8, 9}]

5

Triangles:

[{0, 1, 4}, {3, 5, 6}, {2, 3, 8}, {8, 9, 10}, {4, 7, 10}, {1, 6, 10}]

Triads:

[{0, 4, 7}, {2, 3, 5}, {2, 8, 9}]

6

Triangles:

[{3, 5, 6}, {0, 1, 4}, {2, 3, 8}]

Triads:

[{7, 9, 10}, {0, 1, 10}, {5, 6, 10}, {2, 3, 5}, {0, 4, 7}, {2, 8, 9}]

These are all fan-extensions of $M_{9,2}$, which completes the proof.

Lemma 6. *Every 3-connected matroid in \mathcal{M} with a proper $M_{9,1}$ -minor must have an $M_{9,15}$ - or $M_{9,18}$ -minor.*

```
M1 = cat[9][1]
exList = [M for M in M1.linear_extensions(9, simple=True)
          if is_fragile(M) and not M.has_minor(M15)
          and not M.has_minor(M18)
          and not any(M.has_minor(N) for N in
excluded_minors)]
exList.extend([M for M in M1.linear_coextensions(9, cosimple=True)
              if is_fragile(M) and not M.has_minor(M15)
              and not M.has_minor(M18)
              and not any(M.has_minor(N) for N in
excluded_minors)])
print len(exList)
```

0

Hence all matroids arising from $M_{9,1}$ have already been considered.

Lemma 7. *Every 3-connected matroid in \mathcal{M} with an $M_{9,0}$ -minor but no $M_{8,5}$ -, $M_{9,15}$ -, $M_{9,7}$ -minor is a fan-extension of $M_{9,0}$ with respect to $\mathcal{F} = \{(4, 2, 3, 5, 6, 7, 8)\}$.*

We can exclude $M_{8,5}$ because all of its extensions and coextensions have already been dealt with:

```
[i for i in range(20) if cat[9][i].has_minor(cat[8][5])]
[2, 4, 6, 9, 12, 14, 16, 19]
```

Now we verify the conditions of the Fan Lemma:

```
M0 = cat[9][0]
print_fans(M0)
Triangles:
[{0, 1, 4}, {2, 3, 4}, {3, 5, 6}, {6, 7, 8}, {0, 1, 8}, {0, 4, 8},
{1, 4, 8}]
Triads:
[{2, 3, 5}, {5, 6, 7}]
S = excluded_minors + [cat[8][5], cat[9][5], cat[9][15], cat[9][7],
cat[9][17]]
```

```
exList = [M for M in M0.linear_extensions(9, simple=True)
          if is_fragile(M) and not any(M.has_minor(N) for N in S)]
exList.extend([M for M in M0.linear_coextensions(9, cosimple=True)
              if is_fragile(M) and not any(M.has_minor(N)
                                          for N in S)])
print len(exList)
```


2

```

for M in exList:
    print M.rank()
    print_fans(M)
    print " "

```

5

```

Triangles:
[{3, 5, 6}, {2, 3, 4}, {0, 1, 4}, {6, 7, 8}]
Triads:
[{7, 8, 9}, {0, 1, 9}, {5, 6, 7}, {2, 3, 5}]

```

5

```

Triangles:
[{2, 3, 4}, {3, 5, 6}, {6, 7, 8}, {0, 1, 8}]
Triads:
[{0, 1, 9}, {2, 4, 9}, {5, 6, 7}, {2, 3, 5}]

```

```

doubleList = []
for N in exList:
    doubleList.extend([M for M in N.linear_extensions(10,
                                                         simple=True)
                       if is_fragile(M) and not any(M.has_minor(N)
                                                         for N in S)])
    doubleList.extend([M for M in N.linear_coextensions(10,
                                                         cosimple=True)
                       if is_fragile(M) and not any(M.has_minor(N)
                                                         for N in S)])
print len(doubleList)

```

4

```

for M in doubleList:
    print M.rank()
    print_fans(M)
    print " "

```

5

```

Triangles:
[{3, 5, 6}, {2, 3, 4}, {0, 1, 4}, {6, 7, 8}, {8, 9, 10}, {0, 4, 10},
{1, 4, 10}, {0, 1, 10}]
Triads:
[{5, 6, 7}, {2, 3, 5}, {7, 8, 9}]

```

6

```

Triangles:
[{3, 5, 6}, {2, 3, 4}, {6, 7, 8}]
Triads:
[{0, 9, 10}, {1, 9, 10}, {2, 4, 10}, {0, 1, 10}, {2, 3, 5}, {5, 6,

```

```
7}, {0, 1, 9}, {7, 8, 9}]
```

```
5
```

```
Triangles:
```

```
[{2, 3, 4}, {3, 5, 6}, {6, 7, 8}, {0, 1, 8}, {4, 9, 10}, {0, 8, 10},  
{1, 8, 10}, {0, 1, 10}]
```

```
Triads:
```

```
[{2, 3, 5}, {5, 6, 7}, {2, 4, 9}]
```

```
6
```

```
Triangles:
```

```
[{3, 5, 6}, {2, 3, 4}, {6, 7, 8}]
```

```
Triads:
```

```
[{0, 9, 10}, {1, 9, 10}, {0, 1, 10}, {7, 8, 10}, {2, 3, 5}, {5, 6,  
7}, {2, 4, 9}, {0, 1, 9}]
```

All of these are fan-extensions, which completes the proof of the lemma.

Path sequences

Our final step is the study of the matroids that do have an X_8, Y_8, Y_8^* minor. We show they can be generated by repeated segment-cosegment exchanges on designated elements. By our main theorem, we only have to check this up to 12 elements.

```
def DY(M, segment, parallels=None):
    # First, construct the parallel extension of M
    if parallels is not None:
        for e in parallels:
            M = M.linear_extension(element=len(M), chain={e: 1})

    # Next, find a suitable representation of the segment:
    BS = M.augment([], segment)
    NBS = [e for e in segment if not e in BS]
    B = M.augment(BS)
    A, row_labels, col_labels = M.representation(reduced=True, B=B,
labels=True)

    # construct the Theta matroid
    row_indices = [row_labels.index(e) for e in BS]
    col_indices = [col_labels.index(e) for e in NBS]
    k = len(segment)
    D = Matrix(A.base_ring(), nrows=k, ncols=k)
    D[0,1] = A[row_indices[1],col_indices[0]] /
A[row_indices[0],col_indices[0]]
    D[1,0] = -D[0,1]
```

```

for i in range(k-2):
    D[i+2,0] = A[row_indices[0],col_indices[i]]
    D[i+2,1] = A[row_indices[1],col_indices[i]]
    D[0,i+2] = A[row_indices[0],col_indices[i]]
    D[1,i+2] = A[row_indices[1],col_indices[i]]

# construct the ground set of the generalized parallel connection
tmpE = M.groundset_list()
delset = []
E = copy(row_labels)
for i in row_indices:
    E[i] = newlabel(tmpE)
    tmpE.append(E[i])
    delset.append(E[i])
E.extend(NBS)
F = copy(col_labels)
for i in col_indices:
    F[i] = newlabel(tmpE)
    tmpE.append(F[i])
    delset.append(F[i])
F.extend(BS)
E.extend(F)

# construct the new representation
AA = Matrix(A.base_ring(), nrows=A.nrows() + k - 2,
ncols=A.ncols()+2)
AA.set_block(0,0,A)
AA[row_indices[0], A.ncols()+1] = D[0,1]
AA[row_indices[1], A.ncols()] = D[1,0]
for i in range(k-2):
    AA[A.nrows()+i, A.ncols()] = D[i+2,0]
    AA[A.nrows()+i, A.ncols()+1] = D[i+2,1]
return Matroid(groundset=E,reduced_matrix=AA).delete(delset)

def YD(M, cosegment, series=None):
    return DY(M.dual(), segment=cosegment, parallels=series).dual()

```

The segment and cosegment of X_8 are the following:

```

print X8.rank([0,1,4,6])
print X8.corank([2,3,5,7])
S = [0,1,4,6]
C = [2,3,5,7]

```

2

2

```
print "Deletable: ", [e for e in X8.groundset() if has_U25_or_U35(X8 \
e)]
print "Contractible: ", [e for e in X8.groundset() if has_U25_or_U35(X8
/ e)]
```

```
Deletable: [0, 3, 4, 6]
Contractible: [1, 2, 5, 7]
```

X_8 has an automorphism group whose restriction to $\{0, 4, 6\}$ is S_3 , and whose restriction to $\{2, 5, 7\}$ is S_3 :

```
p1 = {0:4, 1:1, 3:3, 4:6, 6:0, 2:7, 5:2, 7:5}
p2 = {0:0, 1:1, 3:3, 6:4, 4:6, 2:5, 5:2, 7:7}
X8.is_isomorphism(X8,p1) and X8.is_isomorphism(X8,p2)
```

```
True
```

```
def move(M):
    """
    Return all matroids that can be produced from M by one allowable
    move.

    Fan moves are broken up into Delta-Y and Y-Delta moves. We use the
    fact that in
    a Delta-Y move, the original elements from X8 survive as an internal
    triangle
    or triad of the fan.
    """
    S = [0,1,4,6]
    C = [2,3,5,7]
    safe_elts = [1,3]
    fanbase = [[0,4,1], [0,6,1], [4,6,1], [2,5,3], [2,7,3], [5,7,3]]

    # Find all allowable segments
    segs = []
    if M.rank(S) == 2:
        segs.append(S)
    if M.rank(C) == 2:
        segs.append(C)
    # add allowable triangles
    for T in fanbase:
        if M.rank(T) == 2:
            segs.append(T)

    # Find all allowable cosegments
    cosegs = []
    if M.corank(S) == 2:
```

```

        cosegs.append(S)
    if M.corank(C) == 2:
        cosegs.append(C)
    # add allowable triads
    for T in fanbase:
        if M.corank(T) == 2:
            cosegs.append(T)
    L = []

    # Delta-Y moves
    for S in segs:
        Sp = [e for e in S if e not in safe_elts]
        for X in Subsets(Sp):
            if len(S) == 3 or (len(S) == 4 and len(X) >= 1):
                L.append(DY(M,S,X))

    # Y-Delta moves
    for C in cosegs:
        Cp = [e for e in C if e not in safe_elts]
        for X in Subsets(Cp):
            if len(C) == 3 or (len(C) == 4 and len(X) >= 1):
                L.append(YD(M,C,X))

    return L

```

```

def grow(pathmatroids, newpathmatroids, n=None):
    """
    Take each of the matroids in `newpathmatroids` and add all
    pathmatroids that can be obtained through one
    (generalized) Delta-Y move. Throw away results on more than `n`
    elements, if these occur.
    """
    L = []
    for M in newpathmatroids:
        L.extend(move(M))
    npm = []
    for N in L:
        if (n is None or N.size() <= n) and not any(M.is_isomorphic(N)
    for M in pathmatroids):
        pathmatroids.append(N)
        npm.append(N)
    return pathmatroids, npm

```

```
pathmatroids = [X8]
```

```

newpathmatroids = [X8]

pathmatroids, newpathmatroids = grow(pathmatroids, newpathmatroids, 12)
print len(pathmatroids)
print len(newpathmatroids)
print len([M for M in pathmatroids if M.size() == 8])
print len([M for M in pathmatroids if M.size() == 9])
print len([M for M in pathmatroids if M.size() == 10])
print len([M for M in pathmatroids if M.size() == 11])
print len([M for M in pathmatroids if M.size() == 12])

```

```

13
12
3
4
4
2
0

```

```

pathmatroids, newpathmatroids = grow(pathmatroids, newpathmatroids, 12)
print len(pathmatroids)
print len(newpathmatroids)
print len([M for M in pathmatroids if M.size() == 8])
print len([M for M in pathmatroids if M.size() == 9])
print len([M for M in pathmatroids if M.size() == 10])
print len([M for M in pathmatroids if M.size() == 11])
print len([M for M in pathmatroids if M.size() == 12])

```

```

80
67
4
10
26
26
14

```

```

pathmatroids, newpathmatroids = grow(pathmatroids, newpathmatroids, 12)
print len(pathmatroids)
print len(newpathmatroids)
print len([M for M in pathmatroids if M.size() == 8])
print len([M for M in pathmatroids if M.size() == 9])
print len([M for M in pathmatroids if M.size() == 10])
print len([M for M in pathmatroids if M.size() == 11])
print len([M for M in pathmatroids if M.size() == 12])

```

```

405
325
4
18
60

```

130

193

```

pathmatroids, newpathmatroids = grow(pathmatroids, newpathmatroids, 12)
print len(pathmatroids)
print len(newpathmatroids)
print len([M for M in pathmatroids if M.size() == 8])
print len([M for M in pathmatroids if M.size() == 9])
print len([M for M in pathmatroids if M.size() == 10])
print len([M for M in pathmatroids if M.size() == 11])
print len([M for M in pathmatroids if M.size() == 12])

```

958

553

4

20

84

253

597

```

pathmatroids, newpathmatroids = grow(pathmatroids, newpathmatroids, 12)
print len(pathmatroids)
print len(newpathmatroids)
print len([M for M in pathmatroids if M.size() == 8])
print len([M for M in pathmatroids if M.size() == 9])
print len([M for M in pathmatroids if M.size() == 10])
print len([M for M in pathmatroids if M.size() == 11])
print len([M for M in pathmatroids if M.size() == 12])

```

1540

582

4

20

92

348

1076

```

pathmatroids, newpathmatroids = grow(pathmatroids, newpathmatroids, 12)
print len(pathmatroids)
print len(newpathmatroids)
print len([M for M in pathmatroids if M.size() == 8])
print len([M for M in pathmatroids if M.size() == 9])
print len([M for M in pathmatroids if M.size() == 10])
print len([M for M in pathmatroids if M.size() == 11])
print len([M for M in pathmatroids if M.size() == 12])

```

1874

334

4

20

93

376
1381

```
pathmatroids, newpathmatroids = grow(pathmatroids, newpathmatroids, 12)
print len(pathmatroids)
print len(newpathmatroids)
print len([M for M in pathmatroids if M.size() == 8])
print len([M for M in pathmatroids if M.size() == 9])
print len([M for M in pathmatroids if M.size() == 10])
print len([M for M in pathmatroids if M.size() == 11])
print len([M for M in pathmatroids if M.size() == 12])
```

1984
110
4
20
93
380
1487

```
pathmatroids, newpathmatroids = grow(pathmatroids, newpathmatroids, 12)
print len(pathmatroids)
print len(newpathmatroids)
print len([M for M in pathmatroids if M.size() == 8])
print len([M for M in pathmatroids if M.size() == 9])
print len([M for M in pathmatroids if M.size() == 10])
print len([M for M in pathmatroids if M.size() == 11])
print len([M for M in pathmatroids if M.size() == 12])
```

1999
15
4
20
93
380
1502

```
pathmatroids, newpathmatroids = grow(pathmatroids, newpathmatroids, 12)
print len(pathmatroids)
print len(newpathmatroids)
print len([M for M in pathmatroids if M.size() == 8])
print len([M for M in pathmatroids if M.size() == 9])
print len([M for M in pathmatroids if M.size() == 10])
print len([M for M in pathmatroids if M.size() == 11])
print len([M for M in pathmatroids if M.size() == 12])
```

1999
0
4
20
93

380
1502

```
for M in pathmatroids:
    if not any(M.has_minor(N) for N in [X8, Y8, Y8d, cat[8][6]]):
        print "Counterexample: "
        print M
print "Done"
```

Done

```
S = [cat[9][3], cat[9][13], cat[9][4], cat[9][14], cat[9][6], cat[9]
[16]]
T = [F7mm, F7mmd, P8m]
TenEltList = []
for M in S:
    TenEltList.extend(M.linear_extensions(element=len(M), simple=True))
TenEltList = [M for M in TenEltList if is_fragile(M) and not
any(M.has_minor(N) for N in T)]
TenEltList.extend([M.dual() for M in TenEltList])
TenEltList = get_nonisomorphic_matroids(TenEltList)
print len(TenEltList)
```

36

```
for i in range(36):
    M = TenEltList[i]
    if not any(M.is_isomorphic(N) for N in pathmatroids):
        print "Counterexample: "
        print i
print "Done"
```

Done

```
ElevenEltList = []
for M in TenEltList:
    ElevenEltList.extend(M.linear_extensions(element=len(M),
simple=True))
ElevenEltList = [M for M in ElevenEltList if is_fragile(M) and not
any(M.has_minor(N) for N in T)]
ElevenEltList.extend([M.dual() for M in ElevenEltList])
ElevenEltList = get_nonisomorphic_matroids(ElevenEltList)
print len(ElevenEltList)
```

90

```
for i in range(len(ElevenEltList)):
    M = ElevenEltList[i]
    if not any(M.is_isomorphic(N) for N in pathmatroids):
        print "Counterexample: "
        print i
print "Done"
```

Done

```
TwelveEltList = []
for M in ElevenEltList:
    TwelveEltList.extend(M.linear_extensions(element=len(M),
simple=True))
TwelveEltList = [M for M in TwelveEltList if is_fragile(M) and not
any(M.has_minor(N) for N in T)]
TwelveEltList.extend([M.dual() for M in TwelveEltList])
TwelveEltList = get_nonisomorphic_matroids(TwelveEltList)
print len(TwelveEltList)
```

255

```
for i in range(len(TwelveEltList)):
    M = TwelveEltList[i]
    if not any(M.is_isomorphic(N) for N in pathmatroids):
        print "Counterexample: "
        print i
print "Done"
```

Done

DEPARTMENT OF MATHEMATICS, LOUISIANA STATE UNIVERSITY, BATON ROUGE,
LOUISIANA, USA

E-mail address: `bclark@lsu.edu, oxley@math.lsu.edu, svanzwam@math.lsu.edu`